

SO3P1179

日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日
Date of Application: 2002年10月 1日

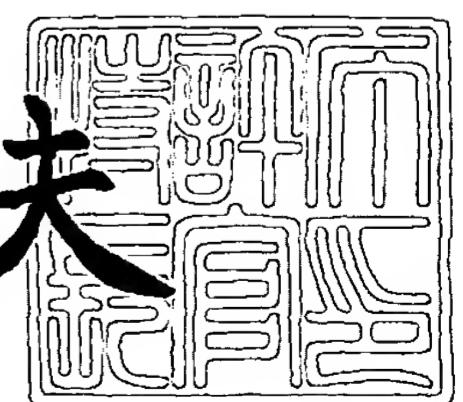
出願番号
Application Number: 特願2002-289022
[ST. 10/C]: [JP2002-289022]

出願人
Applicant(s): ソニー株式会社

2003年 8月11日

特許庁長官
Commissioner,
Japan Patent Office

今井康夫



【書類名】 特許願
【整理番号】 0290283704
【提出日】 平成14年10月 1日
【あて先】 特許庁長官殿
【国際特許分類】 G06F 7/00
【発明者】
【住所又は居所】 東京都品川区北品川6丁目7番35号 ソニー株式会社
内
【氏名】 住広 博
【特許出願人】
【識別番号】 000002185
【氏名又は名称】 ソニー株式会社
【代理人】
【識別番号】 100082131
【弁理士】
【氏名又は名称】 稲本 義雄
【電話番号】 03-3369-6479
【手数料の表示】
【予納台帳番号】 032089
【納付金額】 21,000円
【提出物件の目録】
【物件名】 明細書 1
【物件名】 図面 1
【物件名】 要約書 1
【包括委任状番号】 9708842
【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 情報処理装置および方法、記録媒体、並びにプログラム

【特許請求の範囲】

【請求項1】 複数のモジュール毎にバスの取得を要求するリクエスト信号を受信する複数の受信手段と、

前記複数の受信手段毎に受信されたリクエスト信号に基づいて、前記複数のモジュール毎の待ち時間を計測する計測手段と、

前記計測手段により計測された待ち時間に応じて、前記複数のモジュールのバス取得の優先順位を決定する優先順位決定手段と、

前記優先順位決定手段により決定された優先順位に基づいて、前記複数のモジュールにバスを取得するように制御する制御手段と
を含むことを特徴とする情報処理装置。

【請求項2】 前記計測手段により計測された前記待ち時間が同一である複数のモジュールが存在する場合、前記優先順位決定手段は、前記優先順位をラウンドロビン方式により決定する
ことを特徴とする請求項1に記載の情報処理装置。

【請求項3】 複数のモジュール毎にバスの取得を要求するリクエスト信号を受信する複数の受信ステップと、

前記複数の受信ステップの処理毎に受信されたリクエスト信号に基づいて、前記複数のモジュール毎の待ち時間を計測する計測ステップと、

前記計測ステップの処理により計測された待ち時間に応じて、前記複数のモジュールのバス取得の優先順位を決定する優先順位決定ステップと、

前記優先順位決定ステップの処理で決定された優先順位に基づいて、前記複数のモジュールにバスを取得するように制御する制御ステップと
を含むことを特徴とする情報処理方法。

【請求項4】 複数のモジュール毎に受信されるバスの取得を要求するリクエスト信号に基づいて、前記複数のモジュール毎の待ち時間を計測する計測ステップと、

前記計測ステップの処理により計測された待ち時間に応じて、前記複数のモジ

ュールのバス取得の優先順位を決定する優先順位決定ステップと、
前記優先順位決定ステップの処理で決定された優先順位に基づいて、前記複数
のモジュールにバスを取得するように制御する制御ステップと
を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録され
ている記録媒体。

【請求項5】 複数のモジュール毎に受信されるバスの取得を要求するリク
エスト信号に基いて、前記複数のモジュール毎の待ち時間を計測する計測ステ
ップと、

前記計測ステップの処理により計測された待ち時間に応じて、前記複数のモジ
ュールのバス取得の優先順位を決定する優先順位決定ステップと、

前記優先順位決定ステップの処理で決定された優先順位に基いて、前記複数
のモジュールにバスを取得するように制御する制御ステップと
を含むことを特徴とする処理をコンピュータに実行させるプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、情報処理装置および方法、記録媒体、並びにプログラムに関し、特
に、複数のモジュールが1つのバスを利用した処理をする際に、そのバスを複数
のモジュール間で効率よく取得できるように調停する情報処理装置および方法、
記録媒体、並びにプログラムに関する。

【0002】

【従来の技術】

大規模集積回路 (LSI : Large Scale Integration) により制御される電子機器
が一般に普及してきている。

【0003】

LSIにおいては、複数のIC (Integrated Circuit) からなるモジュール (エン
ジン) が様々な処理を実行する。この際、LSIの各モジュールが処理を実行する
上で、処理に応じて外部メモリに対してデータを記憶させ、または、記憶させた
データを適宜読み出す処理が必要となる。

【0004】

このため、モジュール毎に、外部メモリと接続されたバスの取得順序を調停する、いわゆる、バスアービトレーションを制御する機能が設けられており、各モジュールのバスの取得を制御している。

【0005】

バスアービトレーションの方法としては、ラウンドロビン方式や、固定プライオリティ（固定優先順位）方式が採用されている。

【0006】

ラウンドロビン方式とは、複数のモジュールに対して均等にバスを取得できるようにさせる方式であり、換言すれば、複数のモジュールが順番にバスを取得できるようにするように制御する方式である。

【0007】

図1A、Bは、第1モジュールと第2モジュールの2個のモジュールが、バスを取得するタイミングを、ラウンドロビン方式で制御するときのタイミングチャートの例を示している。

【0008】

ここで、図1Aは、第1モジュールがバスの取得を要求するリクエスト信号を送信するタイミングと、バスを取得して所定の第1の処理を実行するタイミングを示しており、図1Bは、第2モジュールがバスの取得を要求するリクエスト信号を送信するタイミングと、バスを取得して所定の第2の処理を実行するタイミングを示している。

【0009】

第1モジュールは、時刻 t_1 において、バスの取得を要求するリクエスト信号 R_1 を送信し、時刻 t_1 乃至 t_2 において、バスが取得されると、第1の処理1（図中においては、○中に「1」をして表示している：尚、以下、図中の処理を示す表記においては、○中に番号を付すものとする）を実行し、その処理が終了したタイミングの時刻 t_2 において、リクエスト信号 R_2 を送信する。このとき、ラウンドロビン方式では、第1モジュールと第2モジュールが交互にバスが取得されるように制御されるため、第2モジュールが第2の処理 1_1 を完了させ

るまで、第1モジュールはバスを取得することができない。そこで、その次の第1の処理2は、実行されず、待ちの状態となる。

【0010】

ここで、時刻t3において、第2モジュールはリクエスト信号R11を送信し、そのタイミングでは、第2モジュールがバスを取得し、時刻t3乃至t4において、第2の処理11を実行する。

【0011】

この第2の処理11が終了する時刻t4において、第1モジュールはバスを取得することができるので、時刻t4乃至t5において、第1の処理2を実行し、さらに、その第1の処理2が終了する時刻t5において、バスの取得を要求するリクエスト信号R3を送信する。

【0012】

このときも、ラウンドロビン法においては、第1モジュールは、第2モジュールの処理が完了するまでバスを取得することができない。

【0013】

この状態で、時刻t6において、第2モジュールがバス取得のリクエスト信号R12を送信すると、そのタイミングでバスを取得することができるので、時刻t6乃至t7において、第2の処理12を実行する。

【0014】

この処理12が終了する時刻t7において、第1モジュールはバスを取得することができるので、時刻t7乃至t8において、第1の処理3を実行し、さらに、時刻t8において、バスの取得を要求するリクエスト信号R4を送信する。

【0015】

このときも、第1モジュールは、ラウンドロビン法においては、第2モジュールの処理が完了するまでバスを取得することができない。

【0016】

この状態で、時刻t9において、第2モジュールがバス取得のリクエスト信号R13を送信すると、そのタイミングでバスを取得することができるので、時刻t9乃至t10において、第2の処理13を実行し、さらに、このタイミングで

バスの取得を要求するリクエスト信号R14を送信する。このとき、第2モジュールは、ラウンドロビン法において、第1モジュールの処理が完了するまでバスを取得することができない。

【0017】

一方、この第2の処理13が実行されたタイミングである時刻t10において、第1モジュールはバスを取得することができるので、時刻t10乃至t11において、第1の処理4を実行する。このとき、バス取得のリクエストは送信されない。

【0018】

この第1の処理4が終了する時刻t11において、第2モジュールはバスを取得することができるので、時刻t11乃至t12において、第2の処理14を実行し、さらに、時刻t12において、バスの取得を要求するリクエスト信号R15を送信する。このときも、第2モジュールは、ラウンドロビン法において、第1モジュールの処理が完了するまでバスを取得することができない。

【0019】

この状態で、時刻t13において、第1モジュールがバスの取得を要求するリクエスト信号R5を送信すると、そのタイミングでバスを取得することができるので、時刻t13乃至t14において、第1の処理5を実行する。

【0020】

この第1の処理5が終了する時刻t14において、第2モジュールはバスを取得することができるので、時刻t14乃至t15において、第2の処理15を実行し、さらに、時刻t15において、バスの取得を要求するリクエスト信号R16を送信する。このときも、第2モジュールは、ラウンドロビン法において、第1モジュールの処理が完了するまでバスを取得することができない。

【0021】

この状態で、時刻t16において、第1モジュールがバスの取得を要求するリクエスト信号R6を送信すると、そのタイミングでバスを取得することができるので、時刻t16乃至t17において、第1の処理6を実行する。

【0022】

この第1の処理6が終了する時刻t17において、第2モジュールはバスを取得することができるので、時刻t17乃至t18において、第1の処理16を実行する。

【0023】

上述のように、ラウンドロビン方式とは、第1モジュールと第2モジュールが交互にバスを取得して処理を行う方式である。

【0024】

また、固定プライオリティ方式（固定優先順位方式）とは、複数のモジュールが存在する場合に、そのモジュール毎に、優先順位を予め設定し、各モジュールからバスの取得を要求するリクエストがあった場合、その優先順位に応じてバスを取得させる方式である。

【0025】

また、従来のアービトレーション装置には、1つの共有資源を複数のマスタで使用する際に、優先度を算出して算出結果に基づいて共有資源の使用許可を設定するものがある（例えば、特許文献1参照）。

【0026】

【特許文献1】

特開2002-55944号公報（段落[0058]、図3、図4）

【0027】

【発明が解決しようとする課題】

ところで、上述の図1A、Bで示したように、各モジュールの外部メモリへのアクセスの頻度は、処理状態により変化するのが一般的である。

【0028】

しかしながら、例えば、図1A、Bにおいては、例えば、時刻t2、t3、t4において、第1モジュールがバスの取得を要求するリクエスト信号R2、R3、R4を送信しているのに対して、第2モジュールがバスの取得を要求していないにもかかわらず、第2モジュールの処理が終わるタイミングである時刻t4、t7、t10まで、その処理を待たなければならず、時刻t12、t15において、第2モジュールがバスを取得するリクエストR15、R16を送信している

のに対して、第1モジュールがバスの取得を要求していないにもかかわらず、第1モジュールの処理が終わるタイミングである時刻 t_{14} , t_{17} まで、その処理を待たなければならぬため、効率のよいバスアービトレーション処理（バスの取得を調整する処理）が実現できず、結果として無駄な待ち時間が発生し、処理速度がLSI全体として低減してしまうという課題があった。

【0029】

また、固定優先順位方式においては、多くのモジュールが存在する場合、優先順位の高いモジュールは、優先的にバスを取得して処理を実行させることができ、優先順位の低いモジュールは、優先順位の高い処理が実行され続けている限り、バスを取得することができないことになってしまうため、モジュール毎の処理状態が変化したときに適正なバスの取得ができない恐れがあり、結果として処理速度を低下させてしまう恐れがあるという課題があった。

【0030】

本発明はこのような状況に鑑みてなされたものであり、優先順位を変化させて適正なバスアービトレーション処理を実現させるものである。

【0031】

【課題を解決するための手段】

本発明の情報処理装置は、複数のモジュール毎にバスの取得を要求するリクエスト信号を受信する複数の受信手段と、複数の受信手段毎に受信されたリクエスト信号に基づいて、複数のモジュール毎の待ち時間を計測する計測手段と、計測手段により計測された待ち時間に応じて、複数のモジュールのバス取得の優先順位を決定する優先順位決定手段と、優先順位決定手段により決定された優先順位に基づいて、複数のモジュールにバスを取得するように制御する制御手段とを含むことを特徴とする。

【0032】

前記計測手段により計測された待ち時間が同一である複数のモジュールが存在する場合、優先順位決定手段には、優先順位をラウンドロビン方式により決定するようにすることができる。

【0033】

本発明の情報処理方法は、複数のモジュール毎にバスの取得を要求するリクエスト信号を受信する複数の受信ステップと、複数の受信ステップの処理毎に受信されたリクエスト信号に基づいて、複数のモジュール毎の待ち時間を計測する計測ステップと、計測ステップの処理により計測された待ち時間に応じて、複数のモジュールのバス取得の優先順位を決定する優先順位決定ステップと、優先順位決定ステップの処理で決定された優先順位に基づいて、複数のモジュールにバスを取得するように制御する制御ステップとを含むことを特徴とする。

【0034】

本発明の記録媒体のプログラムは、複数のモジュール毎に受信されるバスの取得を要求するリクエスト信号に基づいて、複数のモジュール毎の待ち時間を計測する計測ステップと、計測ステップの処理により計測された待ち時間に応じて、複数のモジュールのバス取得の優先順位を決定する優先順位決定ステップと、優先順位決定ステップの処理で決定された優先順位に基づいて、複数のモジュールにバスを取得するように制御する制御ステップとを含むことを特徴とする。

【0035】

本発明のプログラムは、複数のモジュール毎に受信されるバスの取得を要求するリクエスト信号に基づいて、複数のモジュール毎の待ち時間を計測する計測ステップと、計測ステップの処理により計測された待ち時間に応じて、複数のモジュールのバス取得の優先順位を決定する優先順位決定ステップと、優先順位決定ステップの処理で決定された優先順位に基づいて、複数のモジュールにバスを取得するように制御する制御ステップとを含むことを特徴とする。

【0036】

本発明の情報処理装置および方法、並びプログラムにおいては、複数のモジュール毎にバスの取得を要求するリクエスト信号が受信され、複数に受信されたリクエスト信号に基づいて、複数のモジュール毎の待ち時間が計測され、計測された待ち時間に応じて、複数のモジュールのバス取得の優先順位が決定され、決定された優先順位に基づいて、複数のモジュールにバスが取得されるように制御される。

【0037】

【発明の実施の形態】

図2は、本発明に係るLSIの一実施の形態の構成を示す図である。

【0038】

本発明におけるLSIは、入力画像をMPEG方式で符号化して出力すると共に、入力されたMPEG方式の符号化信号を復号して画像として出力する処理を同時に実行することができるものである。

【0039】

まず、符号化部1の構成について説明する。符号化部1は、入力画像を符号化し、符号化信号として出力するものである。

【0040】

符号化部1の画像入力部21は、入力されてくる画像を符号化で用いる空間解像度に変換し、Iピクチャ（イントラ画像）、Pピクチャ（前方向予測画像）、および、Bピクチャ（前後方向予測画像）のそれぞれのピクチャタイプに合わせて画面を並べ替えて、動き予測部22、および、符号化処理部23に出力する。より詳細には、画像入力部21は、ピクチャタイプ毎にバス8を介して一旦外部メモリ9に出力して、記憶させた後、これを動き予測部22、および、符号化処理部23が読み出す。このとき、画像入力部21は、符号化制御部25に対してバス8を取得するように要求し、この要求に応じて、符号化制御部25からバス8が取得されたことを示す確認の通知を受信すると、バス8を介して外部メモリ9に対して所定の処理を実行させる。

【0041】

動き予測部22は、Bピクチャ、または、Pピクチャのピクチャタイプとして入力された画像と、その基準となるIピクチャ、または、Pピクチャとの差分を求め、動きベクトルを求めて符号化処理部23に出力する。より詳細には、動き予測部22は、外部メモリ9に入力画像として記憶されている画像と相関のある、Iピクチャ、または、Pピクチャを外部メモリ9から読み出して、比較し、動きベクトルを求めて符号化処理部23が読み出せるように、外部メモリ9に記憶させる。このとき、動き予測部22は、符号化制御部25に対してバス8を取得するように要求し、この要求に応じて、符号化制御部25からバス8が取得されたこ

とを示す確認の通知を受信すると、バス8を介して外部メモリ9に対して所定の処理を実行させる。

【0042】

符号化処理部23は、画像入力部21より入力された入力画像と、動き予測部22より入力された動きベクトルに基づいて、ピクチャタイプに応じて符号化信号を生成してビットストリーム出力部24に出力する。より詳細には、外部メモリ9に記憶されている画像入力部21より入力された入力画像と、動き予測部22により生成された動きベクトルとを用いて、Iピクチャの場合、入力画像となった画像をそのまま符号化し、Pピクチャ、または、Bピクチャの場合、その動きベクトルを符号化することにより符号化信号を生成してビットストリーム出力部24により読み出し可能な情報として、外部メモリ9に記憶させる。このとき、符号化処理部23は、符号化制御部25に対してバス8を取得するように要求し、この要求に応じて、符号化制御部25からバス8が取得されたことを示す確認の通知を受信すると、バス8を介して外部メモリ9に対して所定の処理を実行させる。

【0043】

ビットストリーム出力部24は、符号化処理部23により符号化された符号化信号を外部メモリ9から読み出し、ビットストリームとして符号化出力信号を出力する。このとき、ビットストリーム出力部24は、符号化制御部25に対してバス8を取得するように要求し、この要求に応じて、符号化制御部25からバス8が取得されたことを示す確認の通知を受信すると、バス8を介して外部メモリ9に対して所定の処理を実行させる。

【0044】

符号化制御部25は、符号化部1の全体の動作を制御しており、画像入力部21、動き予測部22、符号化処理部23、またはビットストリーム出力部24からバス8を取得するように要求があった場合、バスの取得を要求するリクエスト信号(Request)を端子25aからカウンタ3の端子3aおよびアビトレーションコントローラ7の端子7eに出力する。また、このリクエスト信号に応じてバス8の取得が認められたとき、アビトレーションコントローラ7の端子7d

から送信されてくる確認信号（Acknowledge）を端子25bで受信すると共に、この確認信号が受信されたとき、画像入力部21、動き予測部22、符号化処理部23、またはビットストリーム出力部24に対して、バス8が取得されたことを示す通知を出力して、各種の処理を実行させる。

【0045】

カウンタ3は、端子3aに符号化部1より入力されるバス8の取得を要求するリクエスト信号を受信すると、所定の時間間隔で値をカウントし、そのカウント値Z1をコンパレータ4-1, 4-2に出力する。また、カウンタ3は、アービトレーションコントローラ7の端子7dから端子3bを介して確認信号が入力されると、それまでのカウントを停止し、カウント値をリセットする。尚、この状態でも、カウンタ3は、カウント値をコンパレータ4-1, 4-2に出力している。

【0046】

コンパレータ4-1, 4-2は、それぞれカウンタ3から入力されたカウント値Z1と、所定の値C1, C2とを比較して、カウント値Z1の方が大きいと判定するとき、Trueの信号をアービトレーションコントローラ7の端子7b, 7cにそれぞれ出力し、それ以外のときFalseの信号をアービトレーションコントローラ7の端子7b, 7cにそれぞれ出力する。尚、所定の値C1, C2は、C1 ≠ C2であるため、例えば、C1 < C2である場合、カウンタ3のカウント値が継続されている状態のとき、まず、カウント値Z1が所定の値C1を越えたタイミングで、コンパレータ4-1からTrueの信号が出力され、さらに、カウント値Z1が所定の値C2を超えたタイミングで、コンパレータ4-2からもTrueの信号が出力されることになる。従って、アービトレーションコントローラ7は、コンパレータ4-1, 4-2から端子7b, 7cに入力される信号のパターン（組み合わせ）により符号化部1がバス8の取得を要求するリクエスト信号が送信されてから、バス8の取得を確認する確認信号が入力されるまでの時間を把握することができる。尚、以下の説明においては、C1 < C2であるものとするが、もちろん、C1 > C2であってもよい。

【0047】

次に、復号部2の構成について説明する。

【0048】

復号部2のビットストリーム入力部42は、入力符号化信号をバス8を介して外部メモリ9に順次記憶させる。このとき、ビットストリーム入力部42は、復号制御部41に対してバス8を取得するように要求し、この要求に応じて、復号制御部41からバス8が取得されたことを示す確認の通知を受信すると、バス8を介して外部メモリ9に対して所定の処理を実行させる。

【0049】

復号処理部43は、ビットストリーム入力部42から順次外部メモリ9に記憶された符号化信号を読み出し、動き補償部44に出力する。このとき、復号処理部43は、一度復号したIピクチャ、または、Pピクチャを外部メモリ9に記憶させた後、PピクチャやBピクチャの予測に用いるため再び外部メモリ9から読出す。また、復号処理部43は、入力符号化信号に含まれている予測モードや動きベクトルの情報を動き補償部44に出力し、この情報に基づいて、動き補償部44により生成された動き補償予測情報を用いて画像を復号する。このとき、復号処理部43は、復号制御部41に対してバス8を取得するように要求し、この要求に応じて、復号制御部41からバス8が取得されたことを示す確認の通知を受信すると、バス8を介して外部メモリ9に対して所定の処理を実行させる。

【0050】

動き補償部44は、復号処理部43から入力される復号された画像、動きベクトル、および、予測モードの情報に基づいて、予測モードが動き補償予測モードであるとき、動き補償予測情報を生成して復号された画像に動き補償予測を行い、復号処理部43に出力すると共に、復号処理部43から入力された画像を画像出力部45に出力する。より詳細には、復号処理部43から入力された画像を一旦バス8を介して外部メモリ9に記憶させて、外部出力部45から読み出せるよう記憶させる。このとき、動き補償部44は、復号制御部41に対してバス8を取得するように要求し、この要求に応じて、復号制御部41からバス8が取得されたことを示す確認の通知を受信すると、バス8を介して外部メモリ9に対して所定の処理を実行させる。

【0051】

画像出力部45は、バス8を介して外部メモリ9に記憶されている復号された画像を順次読み出して、出力画像として出力する。このとき、画像出力部45は、復号制御部41に対してバス8を取得するように要求し、この要求に応じて、復号制御部41からバス8が取得されたことを示す確認の通知を受信すると、バス8を介して外部メモリ9に対して所定の処理を実行させる。

【0052】

復号制御部41は、復号部2の全体の動作を制御しており、ビットストリーム入力部42、復号処理部43、動き補償部44、または画像出力部45からバス8を取得するように要求があった場合、バスの取得を要求するリクエスト信号(Request)を端子41aからカウンタ5の端子5aおよびアビトレーションコントローラ7の端子7gに出力する。また、このリクエストに応じてバス8の取得が認められたとき、アビトレーションコントローラ7の端子7fから送信されてくる確認信号(Acknowledge)を端子41bで受信すると共に、この確認信号が受信されたとき、ビットストリーム入力部42、復号処理部43、動き補償部44、または画像出力部45に対して、バス8が取得されたことを示す通知を出力して、各種の処理を実行させる。

【0053】

カウンタ5は、カウンタ3と同様のものである。すなわち、カウンタ5は、端子5aに復号部2より入力されるバス8の取得を要求するリクエスト信号を受信すると、所定の時間間隔で値をカウントし、そのカウント値Z2をコンパレータ6-1, 6-2に出力する。また、カウンタ5は、アビトレーションコントローラ7から端子5bを介して確認信号が入力されると、それまでカウントを停止して、カウント値Z2をリセットする。尚、この状態でも、カウンタ3は、カウント値をコンパレータ6-1, 6-2に出力している。

【0054】

コンパレータ6-1, 6-2は、コンパレータ4-1, 4-2と同様のものである。すなわち、コンパレータ6-1, 6-2は、それぞれカウンタ5から入力されたカウント値Z2と、所定の値C1, C2とを比較して、カウント値Z2の

方が大きいと判定するとき、Trueの信号をアービトレーションコントローラ7の端子7h, 7iにそれぞれ出力し、それ以外のときFalseの信号をアービトレーションコントローラ7の端子7h, 7iにそれぞれ出力する。尚、所定の値C1, C2は、C1 ≠ C2であるため、例えば、C1 < C2である場合、カウンタ5のカウント値Z2が継続されている状態のとき、まず、カウント値Z2が所定の値C1を越えたタイミングで、コンパレータ6-1からTrueの信号が出力され、さらに、カウント値Z2が所定の値C2を超えたタイミングで、コンパレータ6-2からTrueの信号が出力されることになる。従って、アービトレーションコントローラ7は、端子7h, 7iに入力されるコンパレータ6-1, 6-2から入力される信号のパターンにより復号部2がバス8の取得を要求するリクエスト信号を送信してから、バス8の取得を確認する確認信号が入力されるまでの時間を把握することができる。

【0055】

アービトレーションコントローラ7は、符号化部1、および、復号部2から端子7e, 7gを介して入力されるバス8の取得を要求するリクエスト信号、並びに、端子7b, 7c, 7h、および7iから入力されるコンパレータ4-1, 4-2, 6-1, 6-2のそれぞれの判定結果に基づいて、符号化部1、および、復号部2のいずれにバス8の取得させるかを調整し、取得を認めるとき、確認信号を端子7d, 7fより出力すると共に、確認信号を送信した符号化部1、または、復号部2のいずれかのモジュールに対してバス8を取得させるようにし、外部メモリ9へのデータの書き込みと、読み出しを可能な状態にする。また、アービトレーションコントローラ7は、メモリ7aを内蔵しており、バス8の取得を認めたモジュールの情報を記憶し、ラウンドロビン方式によりバス8の取得を決定する際にその情報を利用し、直前にバス8を取得したタイミングが最も古いモジュールがバス8を取得できるように調整する。すなわち、モジュールが、符号化部1と復号部2の2個の場合、直前にバス8を取得したモジュールが符号化部1であったときには、復号部2がバス8を取得できるように調整し、逆に、復号部2が直前にバスを取得していたとき、符号化部1がバス8を取得できるように調整する。

【0056】

尚、図2において、太線はデータの流れを示すものであり、一点鎖線は、符号化部1によるデータをアビトレーションコントローラ7がバス8を介して外部メモリ9に授受する経路を示しており、2点鎖線は、復号部2によるデータをアビトレーションコントローラ7がバス8を介して外部メモリ9に授受する経路を示している。

【0057】

次に、図3のフローチャートを参照して、符号化制御部25による符号化制御処理について説明する。

【0058】

ステップS1において、符号化制御部25は、画像入力部21、動き予測部22、符号化部23、または、ビットストリーム出力部24のいずれかからバス8の取得が要求されているか否かを判定し、画像入力部21、動き予測部22、符号化部23、または、ビットストリーム出力部24のいずれかからバス8の取得が要求されるまでその処理を繰り返す。ステップS1において、画像入力部21、動き予測部22、符号化部23、または、ビットストリーム出力部24のいずれかからバス8の取得が要求された場合、バス8の取得が要求されたと判定され、ステップS2において、符号化制御部25は、端子25aからカウンタ3の端子3a、および、アビトレーションコントローラ7の端子7eにバス8の取得を要求するリクエスト信号を送信する。

【0059】

ステップS3において、符号化制御部25は、アビトレーションコントローラ7の端子7dから自らの端子25bに確認信号が送信されてきたか否かを判定し、確認信号が送信されてくるまで、その処理を繰り返し、例えば、確認信号が送信されてきた、すなわち、バス8の取得が認められたと判定した場合、その処理は、ステップS4に進む。

【0060】

ステップS4において、符号化制御部25は、バス8の取得を要求してきた、画像入力部21、動き予測部22、符号化部23、または、ビットストリーム出

力部24のいずれかに対して、バス8の取得が認められたことを示す信号を送信し、各種の処理を実行させると共に、その処理は、ステップS1に戻り、それ以後の処理が繰り返される。

【0061】

次に、図4のフローチャートを参照して、カウンタ3によるカウント処理について説明する。

【0062】

ステップS11において、カウンタ3は、リクエスト信号が端子3aに入力されたか否かを判定する。例えば、図3のフローチャートを参照して説明したステップS2の処理により、リクエスト信号が送信されてきた場合、リクエスト信号が端子3aに入力されたと判定し、ステップS12において、カウンタ3は、カウントを開始する。

【0063】

ステップS13において、カウンタ3は、今現在カウントしているカウント値Z1をコンパレータ4-1, 4-2に出力する。

【0064】

ステップS14において、カウンタ3は、端子3bにアビトリエーションコントローラ7の端子7dから確認信号が入力されたか否かを判定し、確認信号が入力されていないと判定した場合、その処理は、ステップS13に戻り、再び、カウント中のカウント値Z1をコンパレータ4-1, 4-2に出力する。すなわち、確認信号が入力されるまで、ステップS13, S14の処理が繰り替えされ、時々刻々と変化して、カウントされていくカウント値Z1がコンパレータ4-1, 4-2に出力され続ける。

【0065】

ステップS14において、アビトリエーションコントローラ7の端子7dから確認信号が入力されたと判定された場合、ステップS15において、カウンタ3は、これまで続けていたカウントを停止すると共に、カウント値Z1をクリアして（Z1=0にして）出力し、その処理は、ステップS11に戻る。

【0066】

すなわち、カウンタ3は、符号化部1からリクエスト信号が出力されてから、これに対応してアービトレーションコントローラ7から確認信号が入力されるまでの間、カウント値Z1をカウントアップし続け、確認信号が入力されるとカウントを停止して、カウント値Z1をクリア（リセット）する。

【0067】

次に、図5のフローチャートを参照して、コンパレータ4-1による比較処理について説明する。

【0068】

ステップS21において、コンパレータ4-1は、カウンタ3より入力されるカウント値Z1を取得する。

【0069】

ステップS22において、コンパレータ4-1は、カウンタ3より取得したカウント値Z1と所定の値C1とを比較し、カウント値 $Z_1 > C_1$ であるか否かを判定する。例えば、カウント値Z1のカウントが進んでおらず、 $Z_1 > C_1$ ではないと判定された場合、ステップS23において、コンパレータ4-1は、falseの信号をアービトレーションコントローラ7の端子7bに出力し、その処理は、ステップS21に戻る。

【0070】

ステップS22において、 $Z_1 > C_1$ であると判定された場合、ステップS24において、コンパレータ4-1は、Trueの信号をアービトレーションコントローラ7の端子7bに出力し、その処理は、ステップS21に戻り、それ以降の処理が繰り返される。

【0071】

すなわち、カウンタ3のカウント値Z1が、所定の値C1を超えるまでカウントが進んだとき、すなわち、リクエスト信号が送信されたタイミングから、アービトレーションコントローラ7から確認信号が送信されてくるまでの待ち時間が所定の値C1に対応する時間を超えたとき、アービトレーションコントローラ7の端子7bにTrue信号が入力されることになる。

【0072】

同様にして、コンパレータ4-2においても同様の処理がなされ、True、または、Falseの信号が、アービトレーションコントローラ7の端子7cに入力される。ただし、所定の値C1、C2は、 $C1 \neq C2$ であるので、例えば、 $C1 < C2$ の場合、カウント値Z1の値がC1を超えた直後は、コンパレータ4-1からはTrue信号が出力されるが、コンパレータ4-2からはFalse信号が出力されることになる。さらに、待ち時間が進み、カウント値Z1が所定の値C2を超えると、コンパレータ4-1、4-2のいずれからもTrue信号が出力されることになる。結果として、アービトレーションコントローラ7には、カウント値Z1の進度に応じて、コンパレータ4-1、4-2から入力されるTrue、または、falseの組み合わせが変化して入力される。

【0073】

次に、図6のフローチャートを参照して、復号部2の復号制御部4-1による復号制御処理について説明する。

【0074】

ステップS3-1において、復号制御部4-1は、ビットストリーム入力部4-2、復号処理部4-3、動き補償部4-4、または、画像出力部4-5のいずれかからバス8の取得が要求されているか否かを判定し、復号制御部4-1は、ビットストリーム入力部4-2、復号処理部4-3、動き補償部4-4、または、画像出力部4-5のいずれかからバス8の取得が要求されるまでその処理を繰り返す。ステップS3-1において、復号制御部4-1は、ビットストリーム入力部4-2、復号処理部4-3、動き補償部4-4、または、画像出力部4-5のいずれかからバス8の取得が要求された場合、バス8の取得が要求されたと判定され、ステップS3-2において、復号制御部4-1は、端子4-1aからカウンタ5の端子5a、および、アービトレーションコントローラ7の端子7gにバス8の取得を要求するリクエスト信号を送信する。

【0075】

ステップS3-3において、符号化制御部2-5は、アービトレーションコントローラ7の端子7fから自らの端子4-1bに確認信号が送信されてきたか否かを判定し、確認信号が送信されてくるまで、その処理を繰り返し、例えば、確認信号

が送信されてきた、すなわち、バス8の取得が認められたと判定した場合、その処理は、ステップS34に進む。

【0076】

ステップS34において、復号制御部41は、バス8の取得を要求してきたビットストリーム入力部42、復号処理部43、動き補償部44、または、画像出力部45のいずれかに対して、バス8の取得が認められたことを示す信号を送信し、各種の処理を実行させると共に、その処理は、ステップS31に戻り、それ以降の処理が繰り返される。

【0077】

尚、カウンタ5によるカウント処理、および、コンパレータ6-1, 6-2による比較処理は、図4を参照して説明したカウンタ3によるカウント処理、および、図5を参照して説明して比較処理と同様であるので、その説明は省略する。ただし、カウンタ5によるカウント値Z2は、コンパレータ6-1, 6-2にそれぞれ入力され、コンパレータ6-1, 6-2の比較結果となる信号は、アビトレーションコントローラ7の端子7h, 7iに入力される。

【0078】

次に、図7のフローチャートを参照して、アビトレーションコントローラ7による調停処理について説明する。

【0079】

ステップS51において、アビトレーションコントローラ7は、符号化部1のみからリクエスト信号を受信しているか否かを判定する。すなわち、アビトレーションコントローラ7は、端子7eに符号化部1からリクエスト信号が入力され、かつ、復号部2からリクエスト信号が入力されていない状態であるか否かを判定する。例えば、ステップS51において、符号化部1からのみのリクエスト信号はないと判定された場合、その処理は、ステップS52に進む。

【0080】

ステップS52において、アビトレーションコントローラ7は、復号部2のみからリクエスト信号を受信しているか否かを判定する。すなわち、アビトレーションコントローラ7は、端子7gに復号部2からリクエスト信号が入力され

、かつ、符号化部1からリクエスト信号が入力されていない状態であるか否かを判定する。例えば、ステップS52において、復号部2からのみのリクエスト信号はないと判定された場合、その処理は、ステップS53に進む。

【0081】

ステップS53において、アービトレーションコントローラ7は、符号化部1、および、復号部2のいずれからもリクエスト信号が受信されているか否かを判定する。すなわち、アービトレーションコントローラ7は、端子7eに符号化部1からリクエスト信号が入力され、かつ、復号部2からリクエスト信号が入力されている状態であるか否かを判定する。例えば、ステップS53において、符号化部1、および、復号部2のいずれからもリクエスト信号はないと判定された場合、その処理は、ステップS51に戻り、それ以降の処理が繰り返される。リクエスト信号は、符号化部1、および、復号部2のいずれのモジュールからも入力されていないので、その処理は、ステップS51乃至S53の処理を繰り返す。

【0082】

ステップS53において、符号化部1、および、復号部2のいずれのモジュールからもリクエスト信号が入力されていると判定された場合、ステップS54において、アービトレーションコントローラ7は、コンパレータ4-1から端子7bにTrueの信号が入力されているか否かを判定し、例えば、Trueの信号が入力されていると判定した場合、その処理は、ステップS55に進む。

【0083】

ステップS55において、アービトレーションコントローラ7は、コンパレータ4-2から端子7cにTrueの信号が入力されているか否かを判定し、例えば、Trueの信号が入力されていると判定した場合、その処理は、ステップS56に進む。

【0084】

ステップS56において、アービトレーションコントローラ7は、コンパレータ6-1から端子7hにTrueの信号が入力されているか否かを判定し、例えば、Trueの信号が入力されていると判定した場合、その処理は、ステップS57に進む。

【0085】

ステップS57において、アービトレーションコントローラ7は、コンパレタ6-2から端子7iにTrueの信号が入力されているか否かを判定し、例えば、Trueの信号が入力されると判定した場合、その処理は、ステップS58に進む。

【0086】

ステップS58において、アービトレーションコントローラ7は、ラウンドロビン方式によりメモリ7aに記憶されている過去の記録から、最近バス8が取得されていないモジュールを優先して優先順位を設定する。すなわち、ステップS54乃至S58の処理により、符号化部1、および、復号部2は、それぞれ所定の値C2に対応する時間以上にリクエスト信号を出力してから確認信号を待ち続けていることになるので、いずれも待ち時間は同一であると判断し、待ち時間での比較をせずに、過去のバス8の取得状況に応じて優先順位を設定する。

【0087】

すなわち、今の場合、モジュールは、符号化部1と復号部2の2個であるので、直前で、符号化部1にバス8を取得させた場合、復号部2の優先順位が優先され、優先順位の第1位が復号部2に設定され、第2位に符号化部1が設定される。また、逆に、直前に復号部2にバス8を取得させた場合、符号化部1の優先順位が優先的に設定され、優先順位の第1位に符号化部1が設定され、第2位に復号部2が設定される。

【0088】

ステップS59において、アービトレーションコントローラ7は、設定された優先順位の情報に基づいて、最も優先順位の高いモジュール、すなわち、優先順位が第1位のモジュールに対して確認信号を送信し、バス8の取得させる。すなわち、図2の場合、最も優先順位の高いモジュールが符号化部1であった場合、端子7dから符号化部1に対して確認信号を出力する。また、逆に、最も優先順位の高いモジュールが復号部2であった場合、端子7fから復号部2に対して確認信号を出力する。

【0089】

ステップS51において、符号化部1からのみリクエスト信号が入力されないと判定された場合、ステップS60において、符号化部1に優先順位が優先して設定される。すなわち、ステップS51において、符号化部1からのみリクエスト信号が受信されていると言うことは、調停の必要が無く、符号化部1に優先的に優先順位が設定される。

【0090】

ステップS52において、復号部2からのみリクエスト信号が入力されていると判定された場合、ステップS61において、復号部1に優先順位が優先して設定される。すなわち、ステップS52において、復号部2からのみリクエスト信号が受信されていると言うことは、調停の必要が無く、復号部2に優先的に優先順位が設定される。

【0091】

ステップS54において、コンパレータ4-1から端子7bに入力される信号がTrueではないと判定された場合、ステップS62において、アービトレーションコントローラ7は、コンパレータ6-1から端子7gに入力される信号がTrueであるか否かを判定する。例えば、コンパレータ6-1から端子7gに入力される信号がTrueではないと判定された場合、その処理は、ステップS58に進み、ラウンドロビン方式で優先順位が設定される。

【0092】

すなわち、ステップS54, S62において、いずれも判定がNoとなるということは、符号化部1、および、復号部2のいずれもリクエスト信号を出力してから、所定の値C1に対応する待ち時間を経過しておらず、待ち時間が同一であると判定されるため、待ち時間による比較をせずに、ラウンドロビン方式で優先順位が設定される。

【0093】

ステップS62において、コンパレータ6-1から端子7gに入力される信号がTrueであると判定された場合、その処理は、ステップS61に進み、復号部2に優先的に優先順位が設定される。すなわち、ステップS54において、Noで、ステップS62において、Yesであるということは、符号化部1では、リク

エスト信号を送信してから所定の値C1に対応する待ち時間が発生していないことになるが、復号部2では、リクエスト信号を送信してから所定の値C1に対応する待ち時間が発生することになるため、復号部2の方が、符号化部1よりもリクエスト信号を送信してからの待ち時間が長いことになるので、復号部2に対して優先的に優先順位が設定される。

【0094】

ステップS55において、コンパレータ4-2から端子7cに入力される信号がTrueではないと判定された場合、ステップS63において、アービトレーションコントローラ7は、コンパレータ6-1から端子7hに入力される信号がTrueであるか否かを判定し、コンパレータ6-1から端子7hに入力される信号がTrueではないと判定した場合、その処理は、ステップS60に進む。すなわち、ステップS56において、Noで、かつ、ステップS63において、Noであるということは、符号化部1は、所定の値C1に対応する時間以上の待ち時間が発生していることになるが、復号部2は、所定の値C1に対応する時間にも満たない時間しか待ち時間が発生しないことになり、符号化部1の方がリクエスト信号を出力してからより長い待ち時間が発生しているので、ステップS60において、符号化部1に対して優先的に優先順位が設定される。

【0095】

ステップS63において、コンパレータ6-1から端子7hに入力される信号がTrueであると判定した場合、ステップS64において、アービトレーションコントローラ7は、コンパレータ6-2から端子7i入力される信号がTrueであるか否かを判定し、コンパレータ6-2から端子7iに入力される信号がTrueではないと判定した場合、その処理は、ステップS58に進む。すなわち、ステップS55において、Noで、かつ、ステップS64において、Noであるということは、符号化部1、および、復号部2のいずれも、リクエスト信号を送信してから、所定の値C1に対応する時間までの待ち時間が発生し、所定の値C2に対応する時間までの待ち時間は発生していないことになるため、発生している待ち時間は同一であると判定し、待ち時間による比較を行わずに、ステップS58において、ラウンドロビン方式により優先順位が設定される。

【0096】

ステップS56、または、S57において、コンパレータ6-1から端子7hにTrueの信号が入力されていない、または、コンパレータ6-2から端子7iにTrueの信号が入力されていないと判定された場合、その処理は、ステップS60に進み、符号化部1に優先的に優先順位が設定される。すなわち、ステップS56またはS57がNoであるということは、符号化部1は、所定の値C2に対応する以上の待ち時間が発生しているにもかかわらず、符号化部2は、所定の値C1に対応する時間にも満たない待ち時間である（ステップS56がNoの場合）か、または、所定の値C2に対応する時間にも満たない待ち時間しか発生していないことになるので、いずれにおいても、待ち時間は、復号部2よりも符号化部1の方が長いことになるので、符号化部1に優先的に優先順位を設定する。

【0097】

以上の処理により、複数の複数のモジュールが存在する場合でも、リクエスト信号を送信してからの待ち時間に応じて、効率よく優先順位を設定することができる、例えば、図1で示したようなタイミングで符号化部1と復号部2の処理が実行されるようなとき、処理効率を向上させることができる。

【0098】

図8A、Bは、上述の処理により優先順位を設定して、バス8の取得をモジュール毎に調整したときのリクエスト信号の送信タイミングと各モジュールの処理状態を示すタイミングチャートである。

【0099】

尚、図8A、Bにおいては、説明の都合上、図8Aが符号化部1の処理を示し、図8Bが復号部2の処理を示すタイミングチャートであるものとするが、入れ替わっても同様である。また、図8A、Bにおいては、実線で示す処理のタイミングは、ラウンドロビン方式でバスアービトレーションを行った場合のタイミングチャートであり、図1で示したタイミングチャートと同様のものであり、ここでは、第1モジュールが符号化部1を示し、第2モジュールが復号部2に相当する。また、点線で示した処理のタイミングは、図7のフローチャートを参照して説明した方法によりバスアービトレーション処理（バス取得の調停処理）を行っ

た場合のタイミングチャートである。さらに、点線で示す図7のフローチャートに準じた方法で実行されたバスアービトトレーションによりなされた処理やリクエスト信号を送信するタイミングには、「」を付している。

【0100】

すなわち、時刻 t_1 でリクエスト信号 $R_{1'}$ が送信されると、このときは、符号化部1からのリクエストだけである（図7のステップS51がYesの場合）、符号化部1に確認信号が送信されて、処理 $1'$ が実行され、その処理が終了する時刻 t_2 のタイミングで、新たにリクエスト信号 $R_{2'}$ が送信されると、このときも、符号化部1からのリクエストだけである（図7のステップS51がYesの場合）、符号化部1に確認信号が送信されて、処理 $2'$ が実行される。さらに、処理 $2'$ が終了する時刻 t_3 において、符号化部1からリクエスト信号 $R_{3'}$ が送信されると同時に、復号部2がリクエスト信号 $R_{11'}$ を送信する。このとき、いずれも所定の値 C_1 に対応する時間だけリクエスト信号が送信されてから経過していない（ステップS54がNo、かつ、ステップS62がNo）、ラウンドロビン方式で優先順位が設定されて、復号部2に確認信号が送信されて（直前では符号化部1が優先されるように優先順位が設定されているため）、処理 $11'$ が実行される。処理 $11'$ が終了する時刻 t_4 においては、リクエスト信号を送信しているのは、符号化部1だけとなる（ステップS51がYes）、符号化部1に確認信号を送信し、処理 $3'$ が実行される。

【0101】

処理 $3'$ が実行された後のタイミングである時刻 t_5 において、符号化部1がリクエスト信号 $R_{4'}$ を出力すると、そのタイミングでリクエスト信号を出力しているモジュールは、符号化部1のみである（ステップS51がYes）、符号化部1に確認信号が送信されて、処理 $4'$ が実行される。

【0102】

処理 $4'$ が終了する時刻 t_6 において、復号部2は、リクエスト信号 $R_{12'}$ を出力する。このとき、リクエスト信号を送信しているモジュールは、復号部2だけである（ステップS52がYes）ので、復号部2に確認信号が送信されて、処理 $12'$ が実行される。

【0103】

処理4'が終了する時刻t6から2タイミング（時刻t6と時刻t8との時間差）遅れた時刻t8において、符号化部1は、リクエスト信号R5'を出力する（図1においても、処理4の後、2タイミング遅れてリクエスト信号R5が出力されている）。このとき、リクエスト信号を送信しているモジュールは、符号化部1だけである（ステップS51がYes）ので、符号化部1に確認信号が送信されて、処理5'が実行される。

【0104】

処理5'が終了する時刻t9のタイミングで、復号部2は、リクエスト信号R13'を出力する。このとき、リクエスト信号を送信しているモジュールは、復号部2だけである（ステップS52がYes）ので、復号部2に確認信号が送信されて、処理12'が実行される。

【0105】

さらに、処理13'が終了する時刻t10において、復号部2はリクエスト信号R14'を出力する。このときも、リクエスト信号を送信しているモジュールは、復号部2だけである（ステップS52がYes）ので、復号部2に確認信号が送信されて、処理14'が実行される。

【0106】

処理14'が終了する時刻t11において、復号部2は、リクエスト信号R15'を出力する。同時に、符号化部1も、リクエスト信号R6'を出力する。このとき、いずれもリクエスト信号が送信されてから所定の値C1に対応する時間だけ経過していないので（ステップS54がNo、かつ、ステップS62がNo）、ラウンドロビン方式で優先順位が設定されて、符号化部1に確認信号が送信されて（直前では復号部2が優先されるように優先順位が設定されて、処理14'が実行されているため）、処理6'が実行される。処理6'が終了する時刻t12においては、リクエスト信号を送信しているのは、復号部2だけとなるので（ステップS52がYes）、復号部2に確認信号が送信され、処理15'が実行される。さらに、処理15'が終了する時刻t13において、復号部2はリクエスト信号R16'を出力するが、このときも、リクエスト信号を出力してい

るモジュールは、復号部2のみであるので、復号部2に確認信号が送信されて、処理16'が実行されて処理が終了する。

【0107】

結果として、ラウンドロビン方式で優先順位を設定するようにしたときには、図1A, Bの双方の処理を同時に実行すると、その処理時間は、時間t1乃至t18であったのに対して、図7のフローチャートを参照して説明した処理によると、処理時間は、時間t1乃至t14となる。

【0108】

このようにバスアービトリエーションを効率よく行うようにすることができるので、LSI全体として処理時間を短縮することができ、結果的に、処理速度を向上させるようにすることができる。

【0109】

また、図7のフローチャートを参照して説明した優先順位の設定方法は、リクエスト信号が複数のモジュールから入力されていた場合、コンパレータ4-1, 4-2, 6-1, 6-2から端子7b, 7c, 7h, 7iに入力される信号のパターンにより分けられることになる。

【0110】

図9は、これらのパターン毎に設定される優先順位をまとめたものである。尚、図9においては、Trueの信号が「1」、Falseの信号が「0」としてそれぞれ示されている。また、符号化部1をモジュールM0、復号部2をモジュールM1として示している。さらに、図中左列は、優先順位の設定状態を示しており、左から優先順位が第1位、第2位を示しており、R(M0, M1)は、モジュールM0, M1の優先順位をラウンドロビン方式で設定することを示している。

【0111】

図9においては、コンパレータ(Comp)4-1, 4-2, 6-1, 6-2の出力信号が、(0, 0, 0, 0), (1, 0, 1, 0), (1, 1, 1, 1)の場合、ラウンドロビン方式で優先順位が設定され、(1, 0, 0, 0), (1, 1, 0, 0), (1, 1, 1, 0)の場合、モジュールM0が優先順位第1位に設定され、モジュールM1が第2位に設定され、(1, 0, 1, 1), (0, 0,

1, 1), (0, 0, 1, 0) の場合、モジュールM1が優先順位第1位に設定され、モジュールM0が第2位に設定されることが示されている。

【0112】

尚、上述の (*1, *2, *3, *4) は、*1がコンパレータ4-1から出力される信号を、*2がコンパレータ4-2から出力される信号を、*3がコンパレータ6-1から出力される信号を、*4がコンパレータ6-2から出力される信号をそれぞれ示している。

【0113】

従って、アービトレーションコントローラ7は、これらのパターンをテーブルとして予め記憶しておく、このパターンに応じて確認信号を各モジュール、すなわち、今の場合、符号化部1、または、復号部2に送信するようにしてもよい。

【0114】

また、以上の例においては、モジュールとして、符号化部1、および、復号部2の2個から構成されるLSIの構成例について説明してきたが、モジュールの数は、それ以上の数であってもよく、例えば、図10で示すように、モジュールが3個のLSIの構成であってもよい。

【0115】

図10において、モジュール101乃至103は、図2の符号化部1、または、復号部2に対応するものであり、以下においては、モジュールM0乃至M2とも称する。また、カウンタ104乃至106は、図2のカウンタ3または5と対応するものである。さらに、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3は、図2のコンパレータ4-1, 4-2, 6-1、または6-2と対応するものである。ただし、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3は、コンパレータ107-1乃至109-1が所定の値C1とカウント値を比較し、コンパレータ107-2乃至109-2が所定の値C2とカウント値を比較し、コンパレータ107-3乃至109-3が所定の値C3とカウント値を比較しており、ここでは、C1 < C2 < C3であるものとする。また、アービトレーションコントローラ110は、図2のアービトレーションコントローラ7に対

応するものであり、バス111は、バス8に対応するものであり、外部メモリ112は、外部メモリ9に対応するものである。

【0116】

このような構成の場合、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3から出力される信号に対応して、図11乃至図13で示されるようにアービトレイションコントローラ111が優先順位を設定し、バス111を取得させて、外部メモリ112を適宜利用して各種の処理を実行させる。

【0117】

すなわち、図11で示すように、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3からの出力信号が、(0, 0, 0, 0, 0, 0, 0), (1, 0, 0, 1, 0, 0, 1, 0, 0), (1, 1, 0, 1, 1, 0, 1, 1, 0), (1, 1, 1, 1, 1, 1, 1, 1, 1)の場合、モジュールM0乃至M2はラウンドロビン方式により優先順位が設定される。

【0118】

また、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3からの出力信号が、(1, 0, 0, 0, 0, 0, 0, 0), (1, 1, 0, 0, 0, 0, 0, 0), (1, 1, 0, 1, 0, 0, 1, 0, 0), (1, 1, 1, 0, 0, 0, 0, 0, 0), (1, 1, 1, 1, 1, 0, 1, 1, 0)の場合、第1位が、モジュールM0であり、第2位、および、第3位は、モジュールM1, M2からラウンドロビン方式により設定される。

【0119】

さらに、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3からの出力信号が、(0, 0, 0, 1, 0, 0, 1, 0, 0), (0, 0, 0, 1, 1, 0, 1, 1, 0), (0, 0, 0, 1, 1, 1, 1), (1, 0, 0, 1, 1, 0, 1, 1, 0), (1, 0, 0, 1, 1, 1, 1, 1, 0)の場

合、第1位、および、第2位は、モジュールM1、M2からラウンドロビン方式により設定され、第3位が、モジュールM0として設定される。

【0120】

また、コンパレータ107-1乃至107-3、108-1乃至108-3、109-1乃至109-3からの出力信号が、(1, 1, 0, 1, 0, 0, 0, 0), (1, 1, 1, 1, 0, 1, 0, 0), (1, 1, 1, 1, 1, 0, 1, 0), (1, 1, 1, 1, 1, 1, 0, 0)の場合、第1位はモジュールM0、第2位は、モジュールM1、第3位はモジュールM2として設定される。

【0121】

さらに、コンパレータ107-1乃至107-3、108-1乃至108-3、109-1乃至109-3からの出力信号が、(1, 1, 0, 0, 0, 0, 1, 0, 0), (1, 1, 1, 0, 0, 0, 1, 0, 0), (1, 1, 1, 0, 0, 1, 0, 0), (1, 1, 1, 1, 0, 0, 1, 0)の場合、第1位はモジュールM0、第2位は、モジュールM2、第3位はモジュールM1として設定される。

【0122】

また、図12で示すように、コンパレータ107-1乃至107-3、108-1乃至108-3、109-1乃至109-3からの出力信号が、(0, 0, 0, 1, 0, 0, 0, 0, 0), (0, 0, 0, 1, 1, 0, 0, 0, 0), (1, 0, 0, 1, 1, 0, 0, 0, 0), (0, 0, 0, 1, 1, 1, 0, 0, 0), (1, 0, 0, 1, 1, 1, 0, 0, 0), (1, 1, 0, 1, 1, 1, 1, 0, 0)の場合、第1位が、モジュールM1であり、第2位、および、第3位は、モジュールM0、M2からラウンドロビン方式により設定される。

【0123】

さらに、コンパレータ107-1乃至107-3、108-1乃至108-3、109-1乃至109-3からの出力信号が、(1, 0, 0, 0, 0, 0, 1, 0, 0), (1, 1, 0, 0, 0, 0, 1, 1, 0), (1, 1, 1, 0, 0, 0, 1, 1, 0), (0, 1, 1, 1, 1), (1, 1, 0, 1, 0, 0, 1, 1, 0), (1, 1, 1, 0, 0, 1, 1, 0)の場合は、モジュールM1、M2からラウンドロビン方式により設定される。

, 1, 0, 0, 1, 1, 1), (1, 1, 1, 1, 0, 1, 1, 1) の場合、第1位、および、第2位は、モジュールM0, M2からラウンドロビン方式により設定され、第3位が、モジュールM1として設定される。

【0124】

また、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3からの出力信号が、(1, 0, 0, 1, 1, 0, 0, 0), (1, 0, 0, 1, 1, 0, 0, 0), (1, 1, 0, 1, 1, 1, 0, 0) の場合、第1位はモジュールM1、第2位は、モジュールM0、第3位はモジュールM2として設定される。

【0125】

さらに、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3からの出力信号が、(0, 0, 0, 1, 1, 0, 1, 0, 0), (0, 0, 0, 1, 1, 0, 0, 0), (0, 0, 0, 1, 1, 1, 0, 0), (1, 0, 0, 1, 1, 1, 1, 0) の場合、第1位はモジュールM1、第2位は、モジュールM2、第3位はモジュールM0として設定される。

【0126】

また、図13で示すように、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3からの出力信号が、(0, 0, 0, 0, 0, 1, 0, 0, 0), (0, 0, 0, 0, 0, 0, 0, 1, 1, 0), (1, 0, 0, 1, 0, 0, 1, 1, 0), (0, 0, 0, 0, 0, 0, 0, 1, 1, 1), (1, 0, 0, 1, 0, 0, 1, 1, 1), (1, 1, 0, 1, 1, 1, 0, 1, 1) の場合、第1位が、モジュールM2であり、第2位、および、第3位は、モジュールM0, M1からラウンドロビン方式により設定される。

【0127】

さらに、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3からの出力信号が、(1, 0, 0, 1, 0, 0, 0, 0), (1, 1, 0, 1, 1, 0, 0, 0), (1, 1, 1, 1, 1, 1, 1, 1) の場合、

, 1, 0, 0, 0), (1, 1, 0, 1, 1, 0, 1, 0, 0), (1, 1, 1, 1, 1, 0, 0), (1, 1, 1, 1, 1, 1, 1, 1, 0) の場合、第1位、および、第2位は、モジュールM0, M1からラウンドロビン方式により設定され、第3位が、モジュールM2として設定される。

【0128】

また、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3からの出力信号が、(1, 0, 0, 0, 0, 0, 1, 1, 0), (1, 0, 0, 0, 0, 1, 1, 1), (1, 1, 0, 0, 0, 1, 1, 1) の場合、第1位はモジュールM2、第2位は、モジュールM0、第3位はモジュールM1として設定される。

【0129】

さらに、コンパレータ107-1乃至107-3, 108-1乃至108-3, 109-1乃至109-3からの出力信号が、(0, 0, 0, 1, 0, 0, 1, 1, 0), (0, 0, 0, 1, 0, 0, 1, 1), (1, 0, 0, 1, 1, 0, 1, 1) の場合、第1位はモジュールM2、第2位は、モジュールM1、第3位はモジュールM0として設定される。

【0130】

このような処理により、図10においては、リクエスト信号を送信してから確認信号が送信されてくるまでの待ち時間が、所定の値C1に対する待ち時間S、所定の値C2に対応する待ち時間M、所定の値C3に対応する待ち時間Lが設定され、 $C_1 < C_2 < C_3$ であることから $S < M < L$ となるので、各モジュールM0乃至M2毎の待ち時間の組み合わせに応じて、モジュールM0乃至M2の優先順位が設定されて、その中で第1位に設定されているモジュールに対してバス111が取得されるようになり、外部メモリ112に対してデータを記録させると共に読み出せるように調停を行うことができるので、LSI内の各モジュールの処理を全体として効率よく実行させることが可能となり、結果として、処理速度を向上させることが可能となる。

【0131】

尚、以上の例においては、符号化部1と復号部2をモジュールとして設定した場合のLSIの構成例について説明してきたが、モジュールはこれ以外のものであってもよい。また、以上においては、LSIにおける各モジュールを例として説明してきたが、LSI上のモジュールに限らず、例えば、プログラム上の処理を実行する際に、CPUにより処理の優先順位を設定する場合に応用してもよい。

【0132】

さらに、図2の構成例における、画像入力部21、動き予測部22、符号化処理部23、ビットストリーム出力部24、ビットストリーム入力部42、復号処理部43、動き補償部44、および、画像出力部45のそれぞれをモジュールとして構成するようにしてもよい。

【0133】

以上によれば、リクエスト信号を送信してから確認信号が送信されてくるまでの待ち時間の長さに応じて、モジュールが取得するバスの優先順位が設定されるので、適正な優先順位を設定することができ、効率のよいバスアービトレイション（バスの取得を調停する処理）を実現させることが可能となる。

【0134】

上述した一連の処理は、ハードウェアにより実行させることもできるが、ソフトウェアにより実行させることもできる。一連の処理をソフトウェアにより実行させる場合には、そのソフトウェアを構成するプログラムが、専用のハードウェアに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行させることができ、例えば汎用のパーソナルコンピュータなどに記録媒体からインストールされる。

【0135】

図12は、図2で示したLSIを、それぞれソフトウェアにより実現する場合のパーソナルコンピュータの一実施の形態の構成を示している。パーソナルコンピュータのCPU201は、パーソナルコンピュータの動作の全体を制御する。また、CPU201は、バス204および入出力インターフェース205を介してユーザからキーボードやマウスなどからなる入力部206から指令が入力されると、そ

れに対応してROM(Read Only Memory)202に格納されているプログラムを実行する。あるいはまた、CPU201は、ドライブ210に接続された磁気ディスク221、光ディスク222、光磁気ディスク223、または半導体メモリ224から読み出され、記憶部208にインストールされたプログラムを、RAM(Random Access Memory)203にロードして実行し、出力部207が実行結果を出力する。さらに、CPU201は、通信部209を制御して、外部と通信し、データの授受を実行する。

【0136】

プログラムが記録されている記録媒体は、図12に示すように、コンピュータとは別に、ユーザにプログラムを提供するために配布される、プログラムが記録されている磁気ディスク221（フレキシブルディスクを含む）、光ディスク222（CD-ROM(Compact Disc-Read Only Memory), DVD (Digital Versatile Disc)を含む）、光磁気ディスク223（MD (Mini-Disc) を含む）、もしくは半導体メモリ234などよりなるパッケージメディアにより構成されるだけでなく、コンピュータに予め組み込まれた状態でユーザに提供される、プログラムが記録されているROM202や、記憶部208に含まれるハードディスクなどで構成される。

【0137】

尚、本明細書において、記録媒体に記録されるプログラムを記述するステップは、記載された順序に沿って時系列的に行われる処理は、もちろん、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理を含むものである。

【0138】

【発明の効果】

本発明によれば、適正な優先順位を設定することができ、効率のよいバスアービトレーション（バスの取得を調停する処理）を実現させることが可能となる。

【図面の簡単な説明】

【図1】

従来のバスアービトレーションによる複数のモジュールによる処理のタイミン

グを示すタイミングチャートである。

【図2】

本発明を適用したLSIの構成を示すブロック図である。

【図3】

符号化制御処理を説明するフローチャートである。

【図4】

カウント処理を説明するフローチャートである。

【図5】

比較処理を説明するフローチャートである。

【図6】

復号制御処理を説明するフローチャートである。

【図7】

調停処理を説明するフローチャートである。

【図8】

本発明を適用したバスアービトリエーションによる複数のモジュールによる処理のタイミングを示すタイミングチャートである。

【図9】

図2のコンパレータの出力信号のパターンと優先順位の関係を示す図である。

【図10】

モジュールが3個の場合のLSIの構成例を示すブロック図である。

【図11】

図10のコンパレータの出力信号のパターンと優先順位の関係を示す図である

。

【図12】

図10のコンパレータの出力信号のパターンと優先順位の関係を示す図である

。

【図13】

図10のコンパレータの出力信号のパターンと優先順位の関係を示す図である

。

【図14】

媒体を説明する図である。

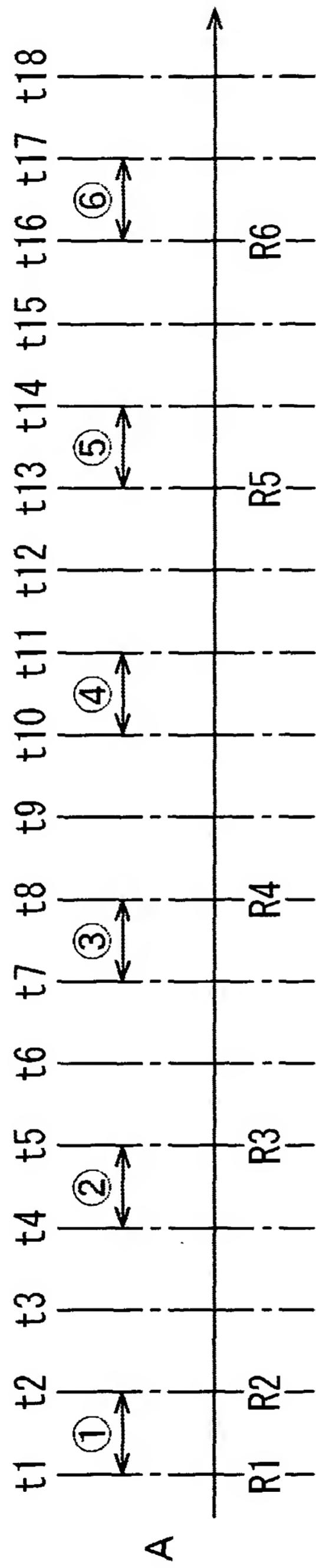
【符号の説明】

1 符号化部, 2 復号部, 3 カウンタ, 3 a, 3 b 端子, 4-1
, 4-2 コンパレータ, 5 カウンタ, 5 a, 5 b 端子, 6-1, 6-
2 コンパレータ, 7 アービトレーションコントローラ, 7 a メモリ,
7 b 乃至 7 i 端子, 8 バス, 9 外部メモリ, 21 画像入力部,
22 動き予測部, 23 符号化処理部, 24 ビットストリーム出力部
, 25 符号化制御部, 25 a, 25 b 端子, 41 復号制御部, 42
ビットストリーム入力部, 43 復号処理部, 44 動き補償部, 45
画像出力部, 101 乃至 103 モジュール, 104 乃至 106 カウン
タ, 107-1 乃至 107-3, 108-1 乃至 108-3, 109-1 乃至
109-3 コンパレータ, 110 アービトレーションコントローラ, 1
11 バス, 112 外部メモリ

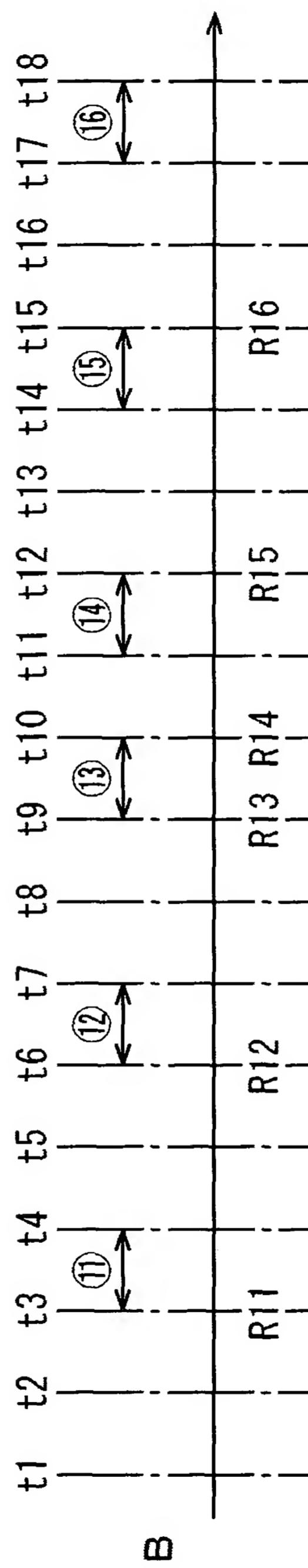
【書類名】 図面

【図 1】

図1

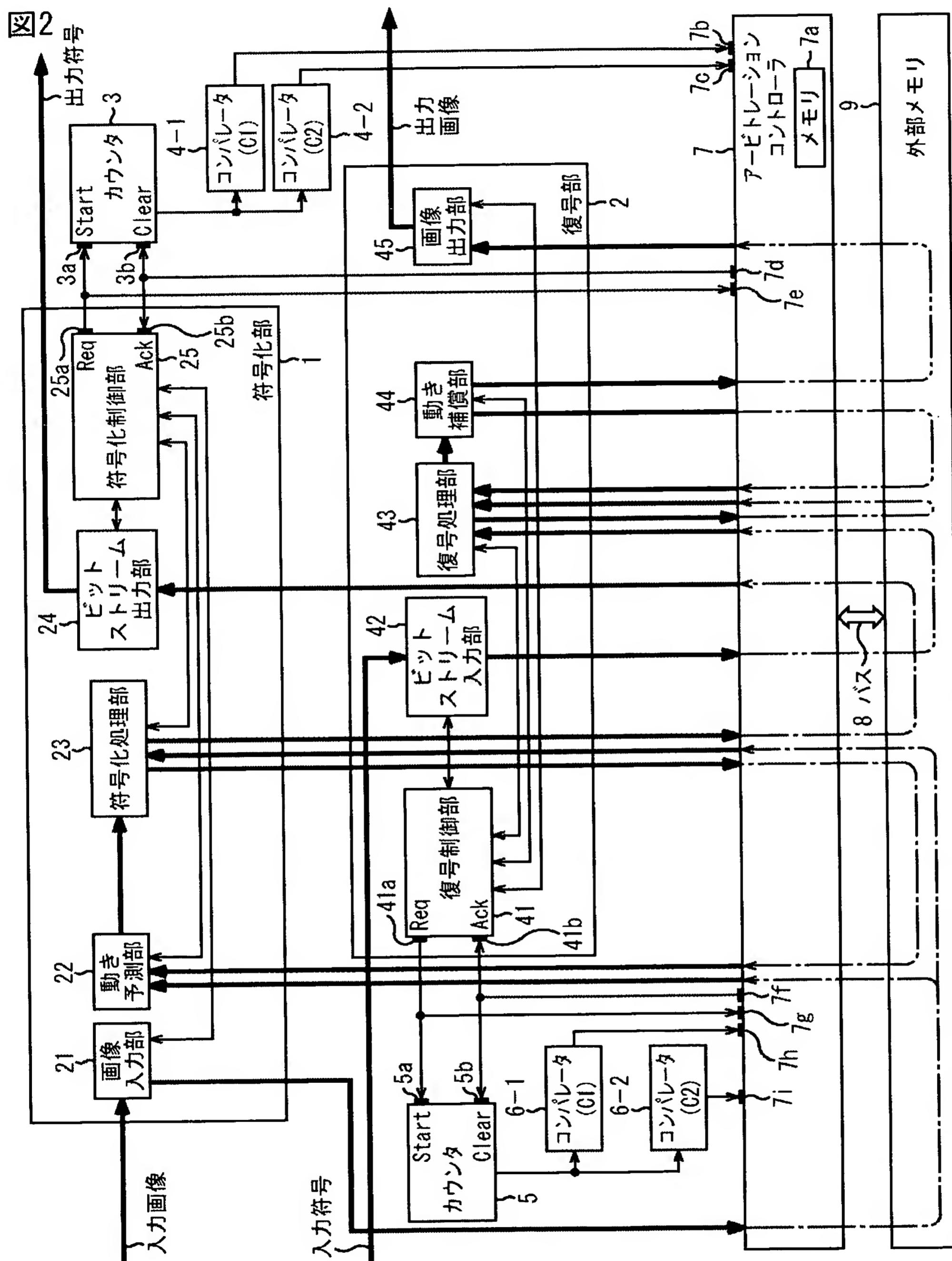


A



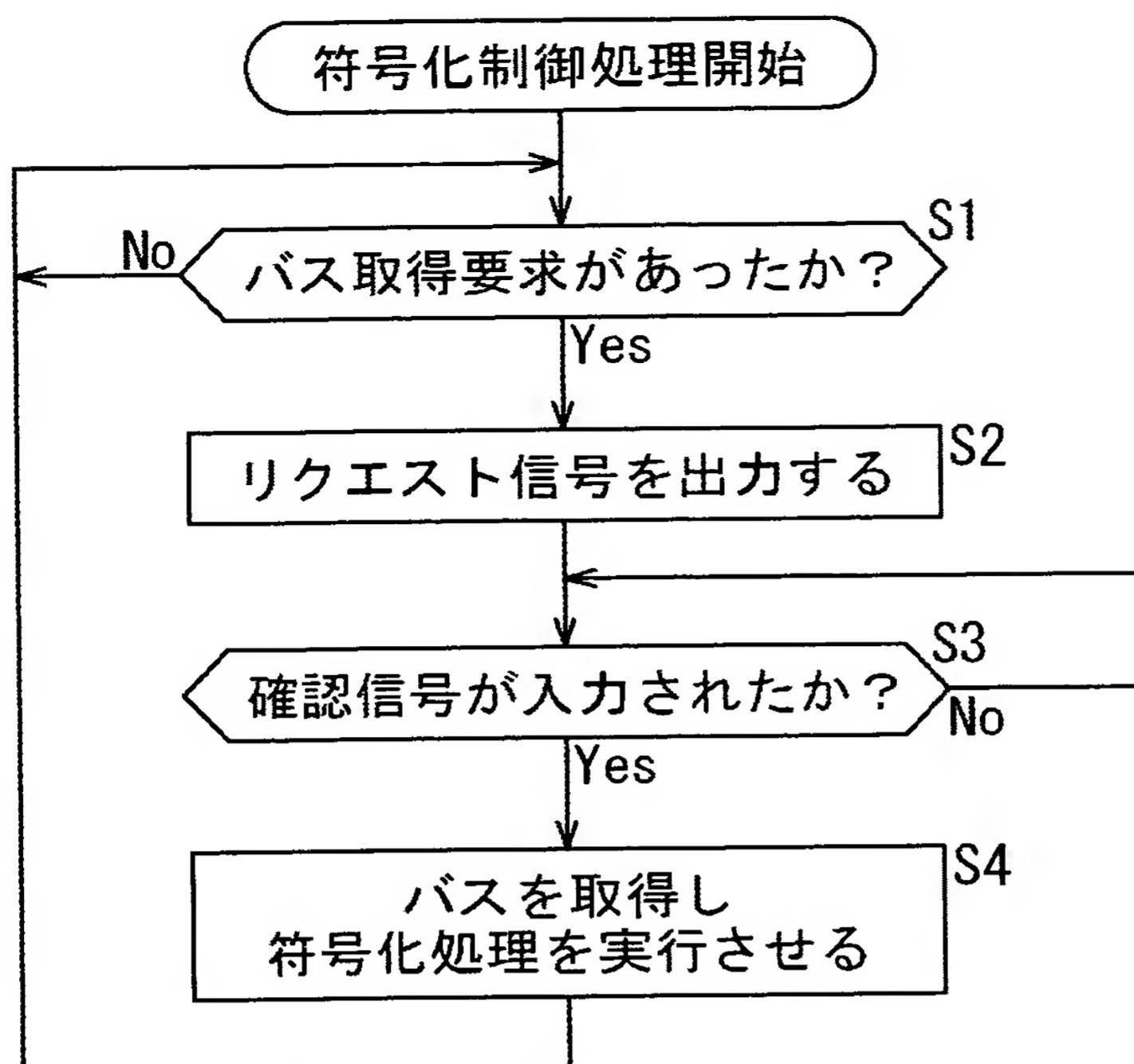
B

図 2】



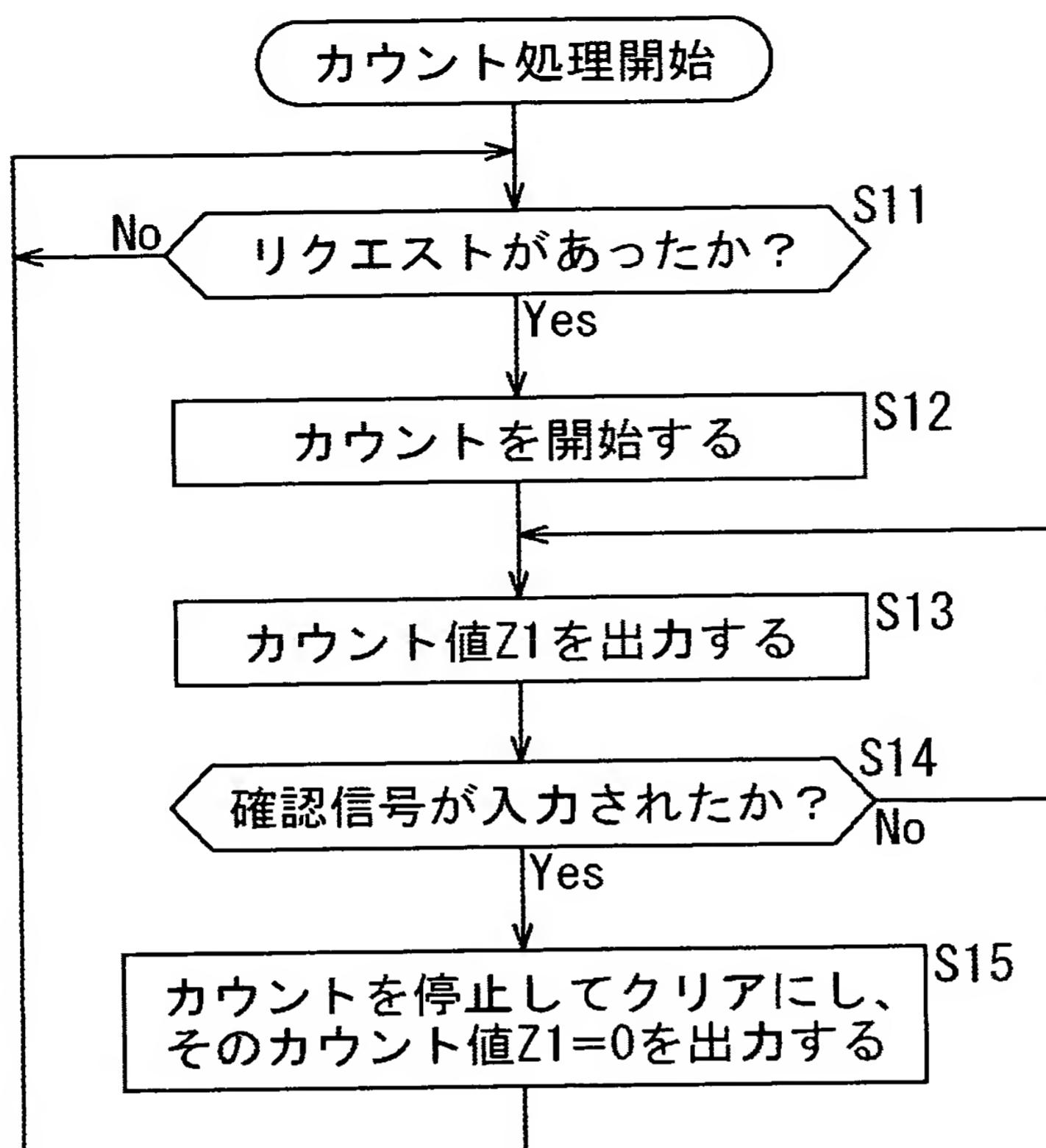
【図3】

図3



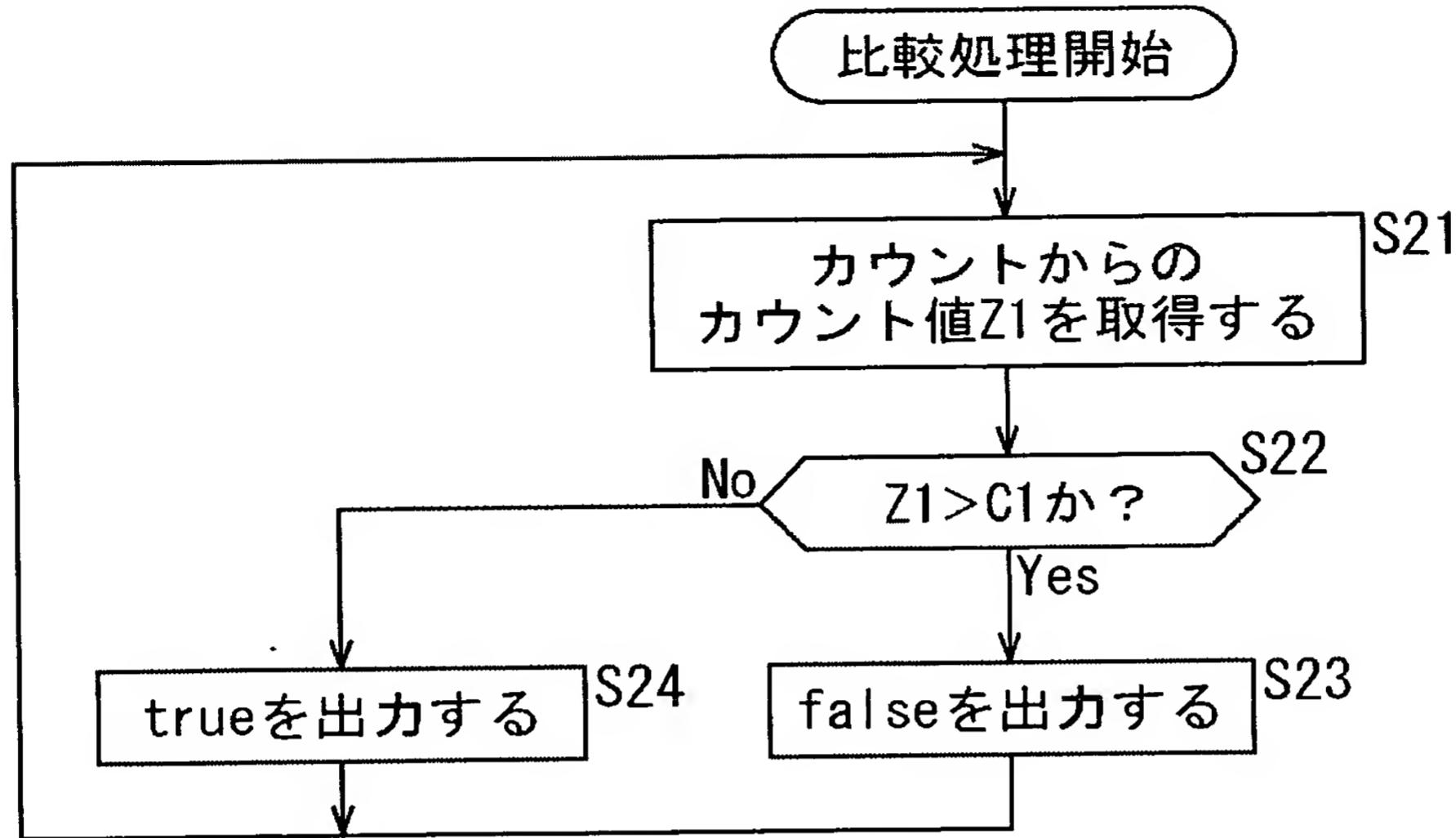
【図4】

図4



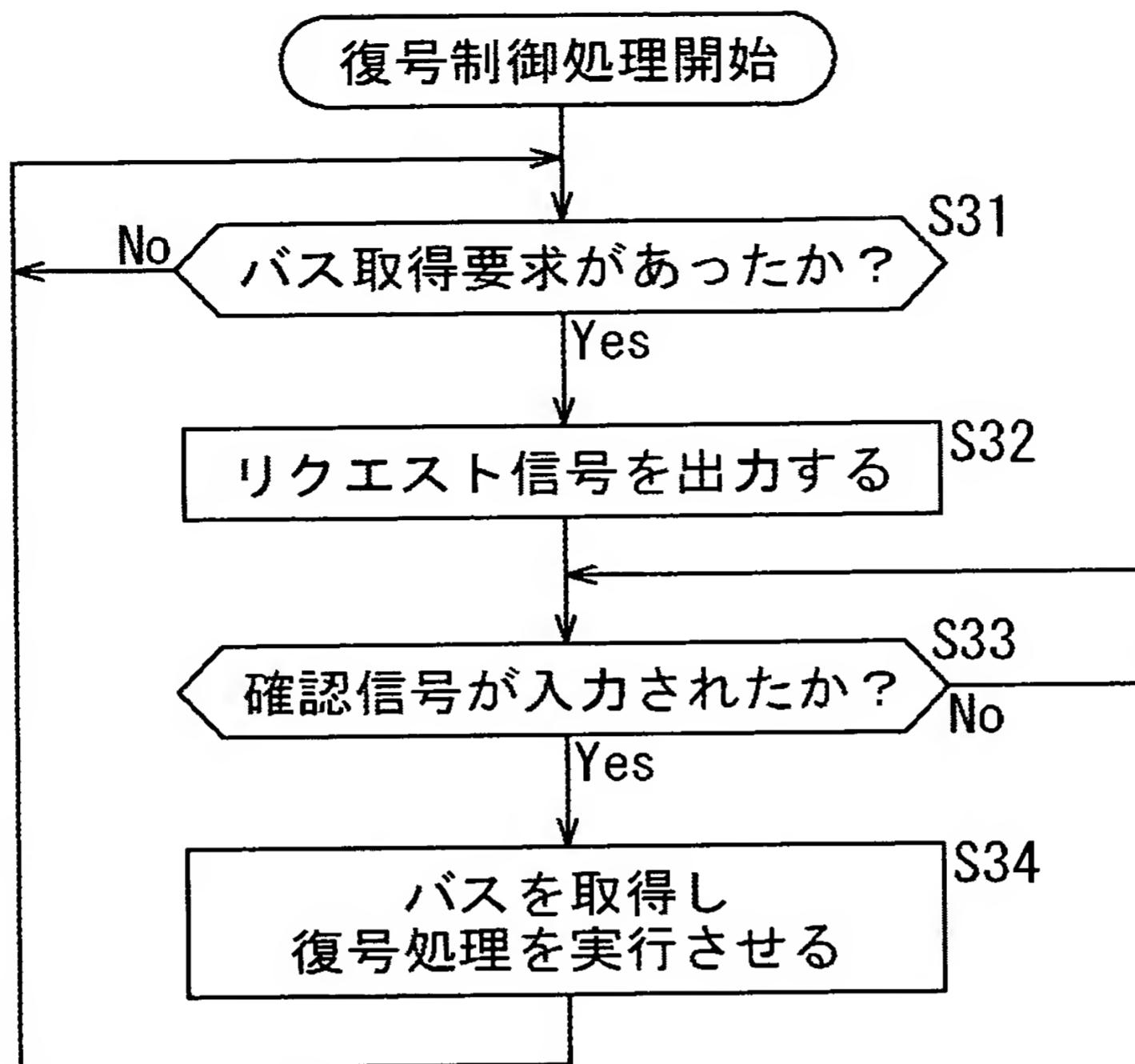
【図 5】

図5

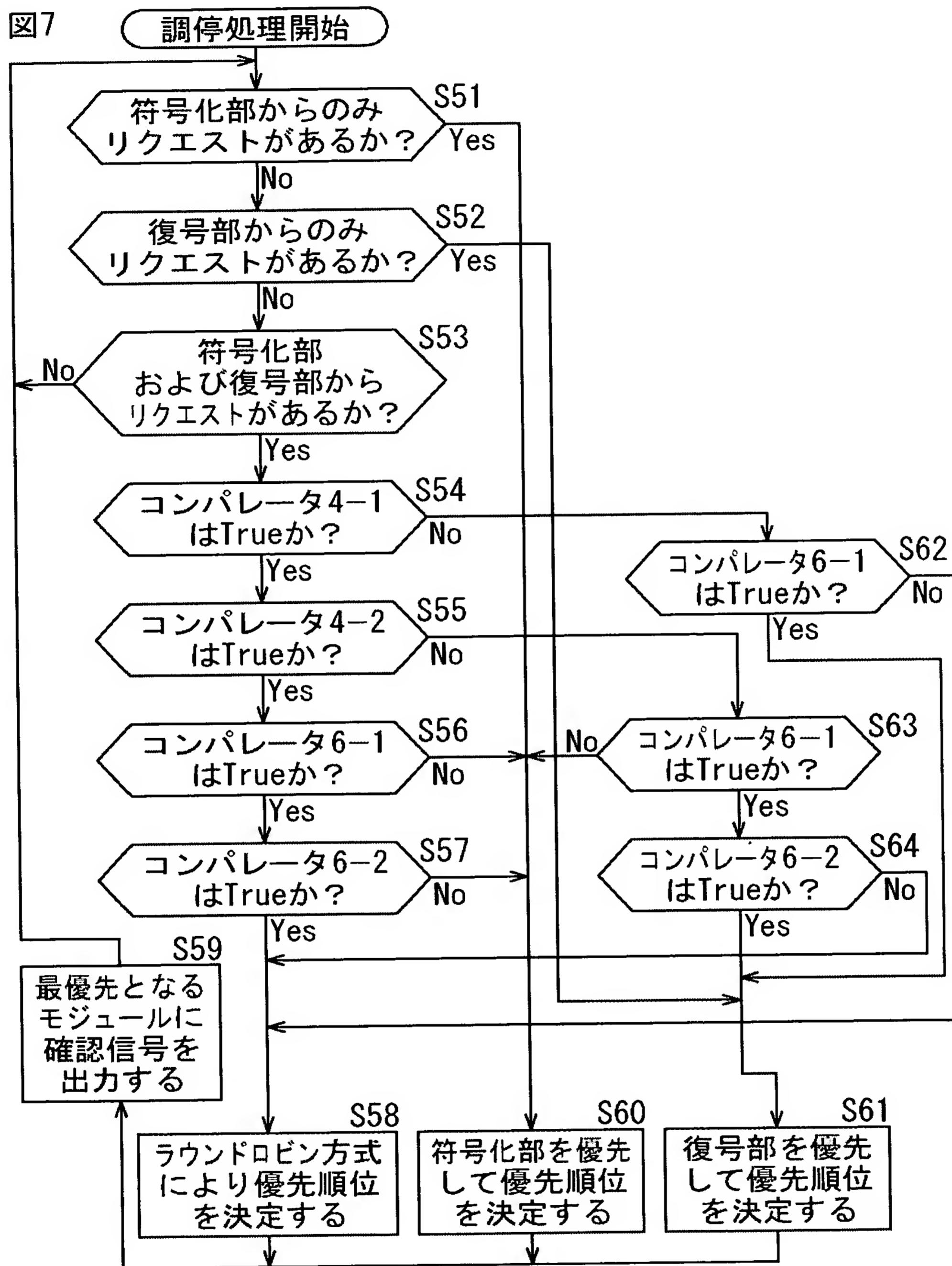


【図6】

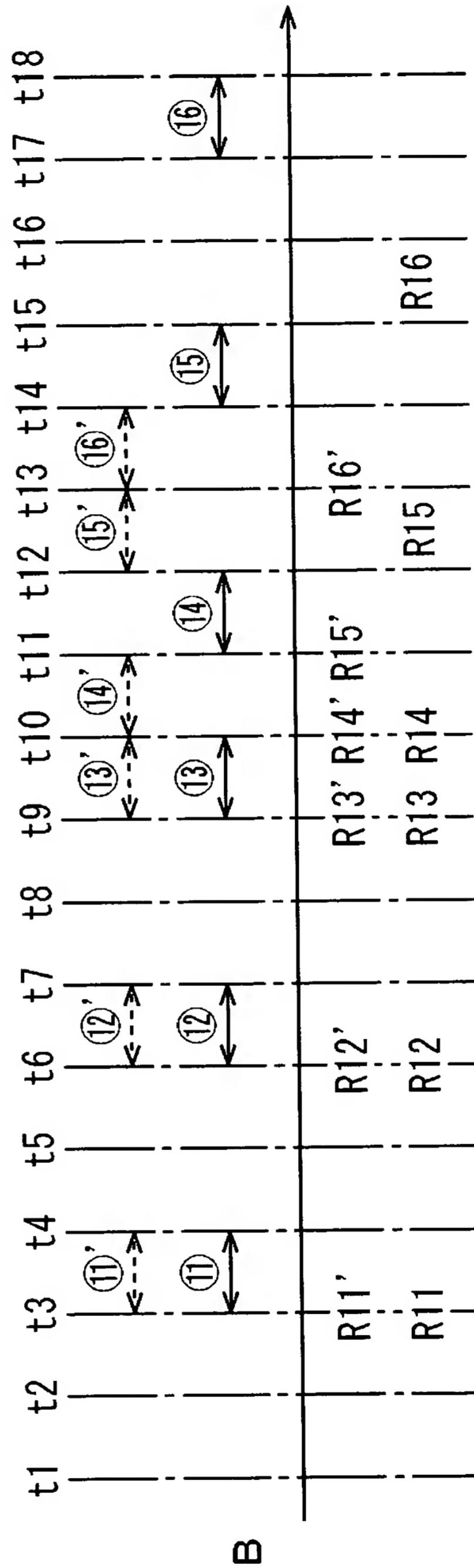
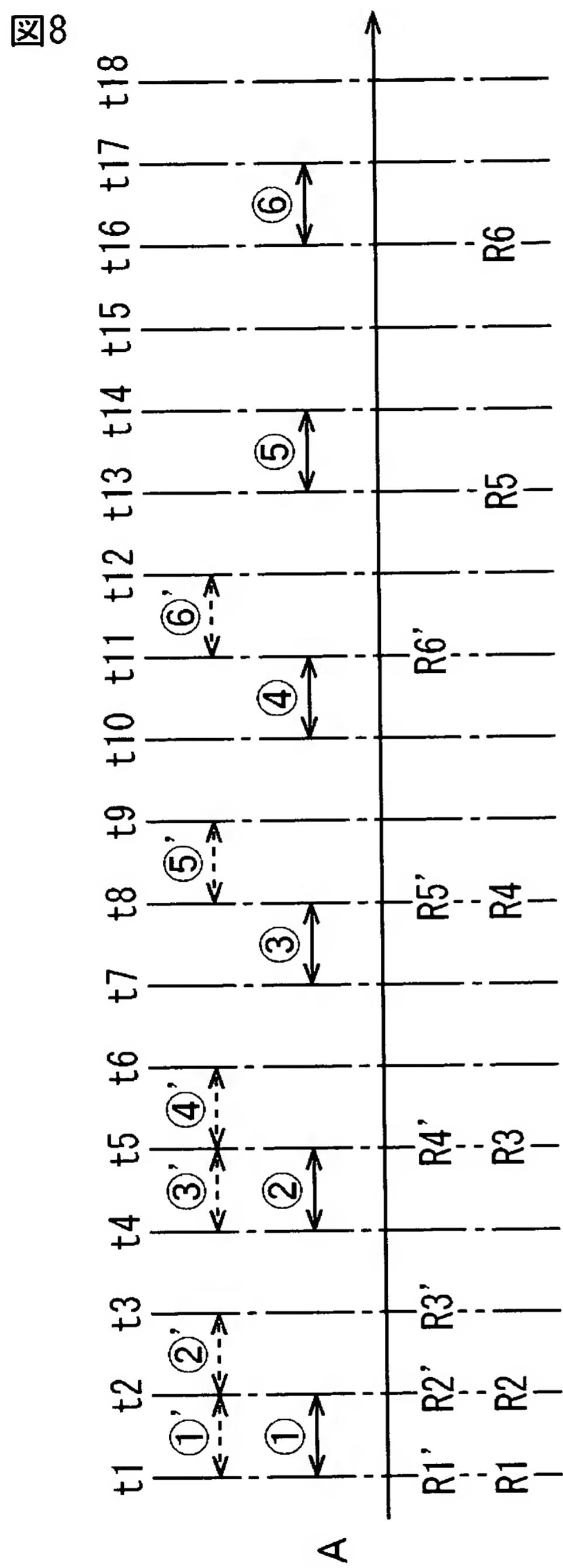
図6



【図7】



【図 8】



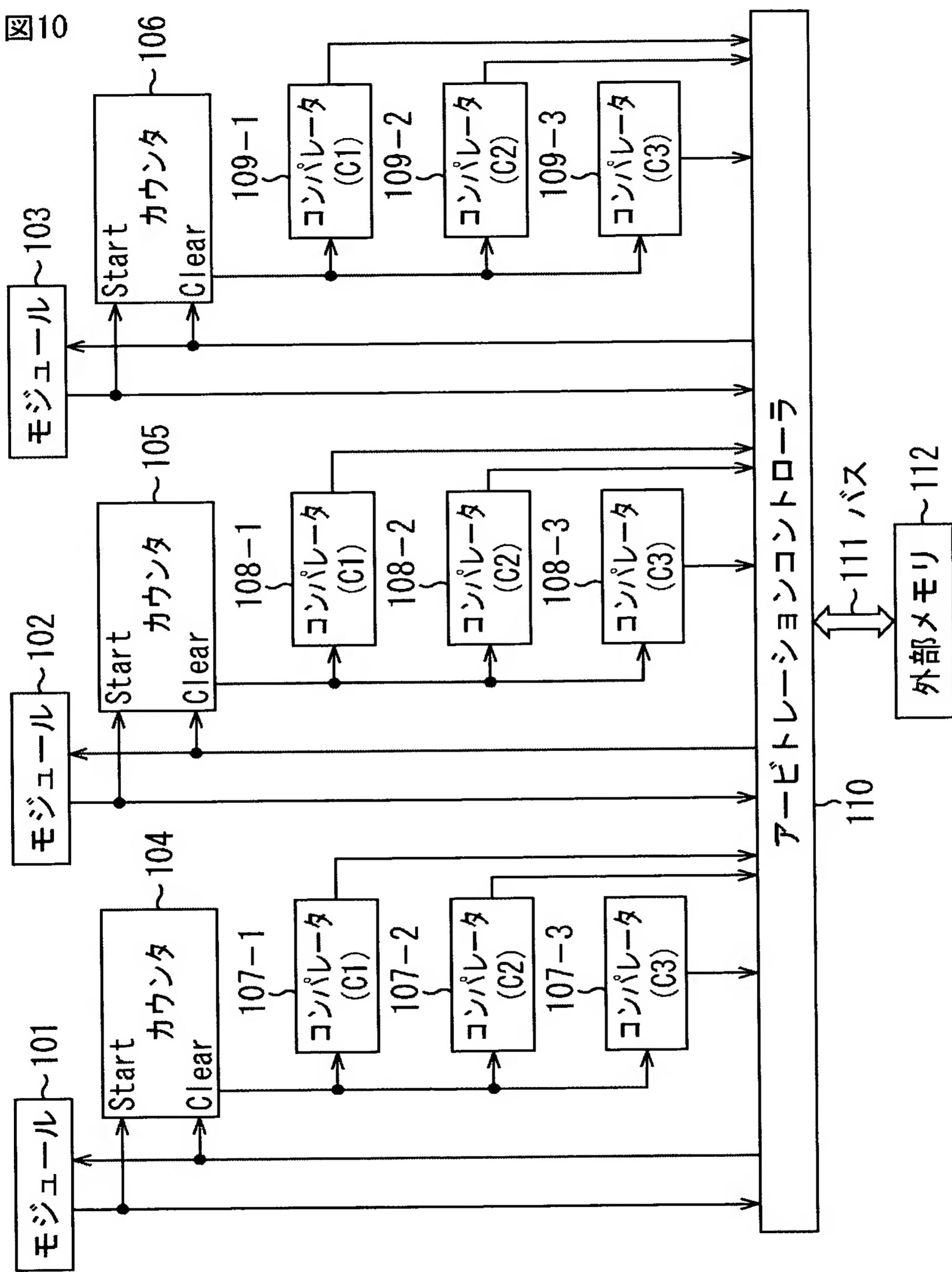
【図9】

図9

	符号化部(M0)		復号部(M1)	
	Comp4-1	Comp4-2	Comp6-1	Comp6-2
R(M0, M1)	0	0	0	0
	1	0	1	0
	1	1	1	1
M0, M1	1	0	0	0
	1	1	0	0
	1	1	1	0
M1, M0	1	0	1	1
	0	0	1	1
	0	0	1	0

【図10】

図10



【図 11】

図 11

	M0			M1			M2		
	Comp107-1	Comp107-2	Comp107-3	Comp108-1	Comp108-2	Comp108-3	Comp109-1	Comp109-2	Comp109-3
R(M0, M1, M2)	0	0	0	0	0	0	0	0	0
	1	0	0	1	0	0	1	0	0
	1	1	0	1	1	0	1	1	0
	1	1	1	1	1	1	1	1	1
M0, R(M1, M2)	1	0	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0	0
	1	0	1	0	0	0	0	0	0
	1	1	0	0	0	0	0	0	0
	1	1	1	0	0	0	1	0	0
	1	1	1	1	0	0	1	1	0
	1	1	1	1	1	0	1	1	0
R(M1, M2), M0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0	0
	1	1	0	0	0	0	0	0	0
	1	1	1	0	0	0	0	0	0
	1	1	1	1	0	0	0	0	0
	1	1	1	1	1	0	0	0	0
M0, M1, M2	1	0	0	1	0	0	0	0	0
	1	1	0	0	1	0	0	0	0
	1	1	1	0	0	1	0	0	0
	1	1	1	1	0	0	1	0	0
	1	1	1	1	1	0	1	0	0
M0, M2, M1	1	0	0	0	0	0	0	0	0
	1	1	0	0	0	0	0	0	0
	1	1	1	0	0	0	0	0	0
	1	1	1	1	0	0	0	0	0
	1	1	1	1	1	0	0	0	0

【図12】

図12

	M0	M1	M2						
	Comp107-1	Comp107-2	Comp107-3	Comp108-1	Comp108-2	Comp108-3	Comp109-1	Comp109-2	Comp109-3
M1, R(M0, M2)	0	0	0	1	0	0	0	0	0
	0	0	0	-1	1	0	0	0	0
	1	0	0	1	0	1	0	0	0
	0	0	0	-1	-1	0	0	0	0
	1	0	0	-1	-1	1	0	0	0
	-1	0	0	1	-1	1	0	0	0
R(M0, M2), M1	0	0	0	0	0	0	0	0	0
	1	0	0	0	0	0	1	1	1
	1	-1	0	0	0	0	-1	-1	-1
	-1	0	0	1	0	0	1	1	1
	1	0	0	1	0	0	1	0	0
	-1	1	1	-1	1	0	-1	-1	-1
	1	-1	-1	-1	-1	1	0	0	0
M1, M0, M2	-1	0	0	0	0	0	0	0	0
	1	0	0	0	1	-1	0	0	0
	1	-1	0	0	1	-1	1	0	0
	-1	1	0	0	1	1	-1	-1	-1
M1, M2, M0	0	0	0	0	-1	0	0	0	0
	0	0	0	0	0	1	1	0	0
	0	0	0	0	1	1	-1	-1	-1
	-1	0	0	0	-1	1	1	1	0

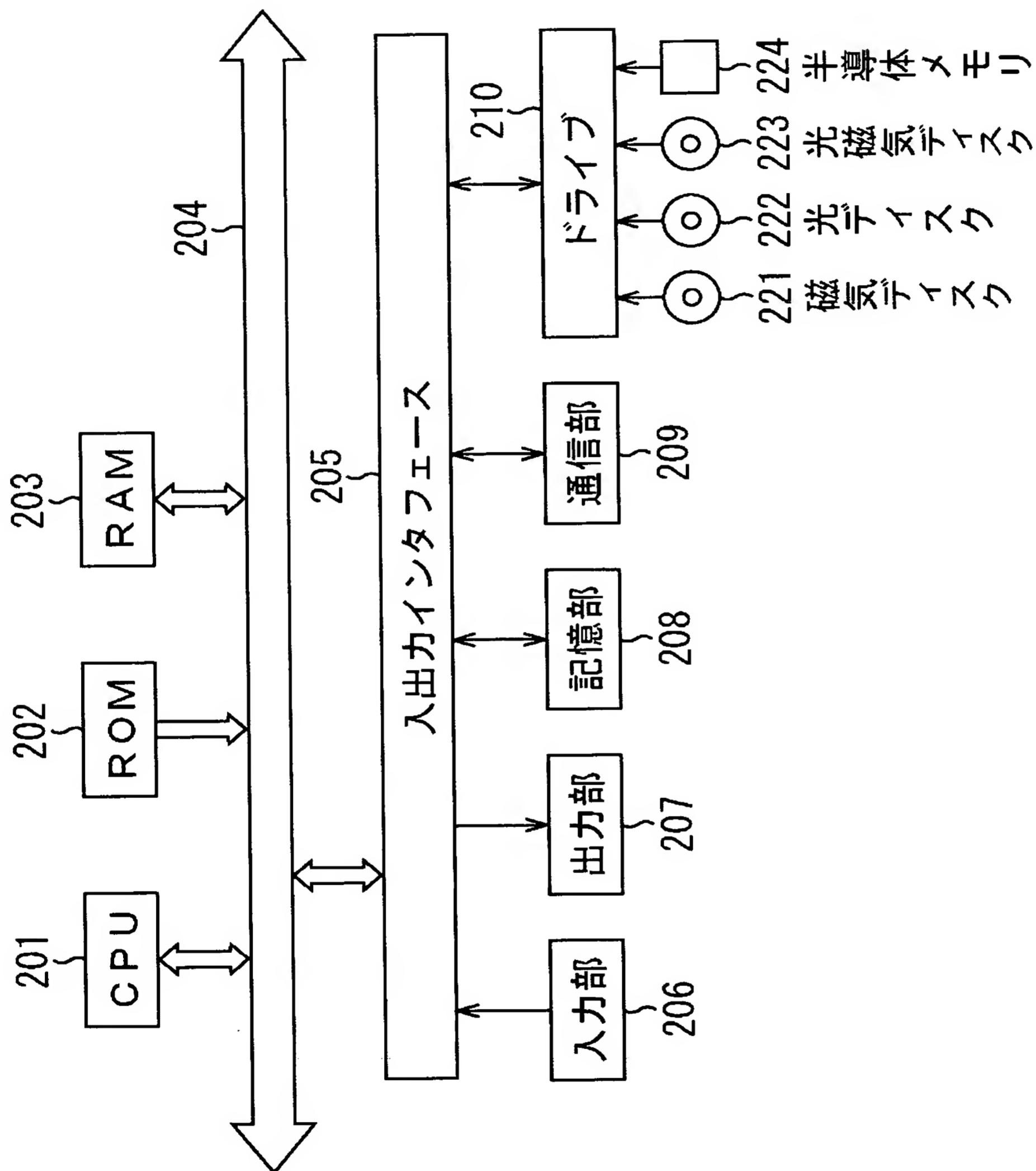
【図13】

図13

	M0			M1			M2		
	Comp107-1	Comp107-2	Comp107-3	Comp108-1	Comp108-2	Comp108-3	Comp109-1	Comp109-2	Comp109-3
M2, R (M0, M1)	0	0	0	0	0	0	1	0	0
	0	0	0	0	0	0	-1	-1	0
	1	0	0	0	0	0	1	1	0
	0	0	0	0	0	0	-1	-1	-1
	1	0	0	0	0	0	1	1	1
	1	1	0	0	1	0	-1	-1	-1
R (M0, M1), M2	-	0	0	-1	0	0	0	0	0
	-	1	0	1	0	0	0	0	0
	-	-	-	-	1	1	0	0	0
	-	-	-	-	1	1	0	0	0
	-	-	-	-	1	0	1	0	0
	-	-	-	-	1	1	1	0	0
M2, M0, M1	-1	0	0	0	0	0	-1	-1	0
	-1	0	0	0	0	0	1	1	1
	-1	1	0	0	0	0	-1	-1	-1
	-1	1	0	0	1	0	1	1	1
M2, M1, M0	0	0	0	1	0	0	-1	-1	0
	0	0	0	0	1	0	1	1	1
	1	0	0	0	1	1	0	1	1
	0	0	0	-1	1	0	1	1	1

【図14】

図14



【書類名】 要約書

【要約】

【課題】 効率のよいバスアービトレーションを実現させる。

【解決手段】 カウンタ3, 5は、それぞれ符号化部1、および、復号部2からリクエスト信号が入力されるとカウントを開始し、カウンタ3がカウンタ値Z1をコンパレータ4-1, 4-2に、カウンタ5がコンパレータ6-1, 6-2にそれぞれ出力する。コンパレータ4-1, 6-1は、所定の値C1と、コンパレータ4-2, 6-2は、所定の値C2と、それぞれカウンタ値Z1, Z2を比較し、比較結果をアービトレーションコントローラ7に出力する。アービトレーションコントローラ7は、このコンパレータ4-1, 4-2, 6-1, 6-2から入力される信号に基いて、符号化部1と復号部2のモジュールの優先順位を決定し、最も優先順位の高いモジュールに確認信号を出力する。本発明は、LSIに適応することができる。

【選択図】 図2

特願2002-289022

出願人履歴情報

識別番号 [000002185]

1. 変更年月日 1990年 8月30日

[変更理由] 新規登録

住 所 東京都品川区北品川6丁目7番35号
氏 名 ソニー株式会社